

Beamforming Techniques for Next Generation Communication Systems

Dissertation

zur Erlangung des akademischen Grades
des Doktor-Ingenieur (Dr.-Ing.)

von: M. Sc. Ricardo Kehrle Miranda

geboren am: 10.12.1985 in Brasília, Brasilien

Matrikelnummer: TU Ilmenau: 56984, UnB: 13/0163783

Gutachter: 1. Univ.-Prof. Dr.-Ing. Giovanni Del Galdo
2. Prof. Dr.-Ing. João Paulo Carvalho Lustosa
da Costa
3. Prof. Dr. Juan José Murillo-Fuentes

eingereicht am: 10.02.2017

Datum des öffentlichen Teils der
wissenschaftlichen Aussprache: 27.03.2017

ACKNOWLEDGMENTS

I would like to thank the following mentors:

My adviser at the University of Brasília, prof. Dr.-Ing. João Paulo Carvalho Lustosa da Costa, for kindly inviting me to be his student and offering a handful of opportunities. Univ.-Prof. Dr.-Ing. Giovanni Del Galdo, my adviser at the TU Ilmenau, for his support and accepting me as a dual degree student at the TU Ilmenau. My co-adviser at both the UnB and the TU Ilmenau Dr.-Ing. Florian Roemer for his patient and detailed comments on this work. Prof. Dr. André Lima Férrer de Almeida for his constructive comments and kind support. Prof. Dr.-Ing. Felix Antreich for having me at the German Aerospace Center and presenting me with new techniques. Prof. Dr. Edison Pignaton de Freitas for his support. Dr. Takashi Hiramatsu, Yuya Nakamura and Takashi Eishima for their mentoring on the development of satellite systems. Prof. Dr. Ricardo Zelenovsky for introducing me to the array signal processing field and prof. Leonardo R.A.X. de Menezes for introducing me to the unscented transformation.

I would like to thank the following sponsors:

The Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) under the PVE grant number 88881.03032/2013-01 for the scholarship number 88887.115692/2016-00 and the Conselho Nacional de Pesquisa e Desenvolvimento Científico e Tecnológico (CNPq) for the scholarship number 232218/2014-5. The Brazilian Space Agency (AEB) for their indication and support. The Fundação de Apoio à Pesquisa do Distrito Federal (FAPDF) for the traveling grants number 193.000.903/2013, 193.000.270/2014 and 193.000.715/2015.

I would like to thank my family, girlfriend, friends and colleagues for their unconditional support, understanding and for the uncountable enjoyable moments.

ABSTRACT

Digital communications, either voice, messaging, video or other media content, have become an essential part of the modern society. As a consequence, the demand for advanced digital communication systems is increasing. Currently, mobile networks have a total of 7.3 billion subscriptions worldwide, from which 1.4 billion belong to the latest fourth generation (4G) network. In 2022, 8.9 billion subscriptions are expected, being 4.3 billion for 4G. Moreover, applications that require a high throughput such as virtual reality (VR) are also foreseen. The communication systems should also fit increasing demands of machine to machine communications, including Internet of things (IoT) and vehicular ad hoc networks (VANETs), such as vehicle to vehicle (V2V) and vehicle to infrastructure (V2I) communications. To support this demand, a 100 fold increase in data rate is being considered as a requirement for future fifth generation (5G) standards, whose deployment starts as early as 2020.

One of the key technologies to allow for a better exploitation of the scarce spectrum is the incorporation of antenna arrays into communication devices. In particular, this work focus on beamforming techniques that can virtually adapt the irradiation pattern of the antenna array based device in order to amplify the signals from a desired direction and to cancel out the interference from other angles. Therefore, beamforming provides the spatial separation of multiple sources sharing the same spectrum band, and can also be applied to mitigate jamming and radio interference.

In this work beamforming techniques and frameworks to deal with colored noise scenarios, uniform rectangular arrays (URA) and broadband scenarios are developed. For colored noise scenarios, prewhitening techniques, rank reduction techniques and a transformation are used. For the URA and the broadband scenarios, a tensor notation is adopted and the parallel factor analysis (PARAFAC) tensor decomposition is used along with frequency invariant beamformers (FIBs). Finally, a low computational cost evaluation method that uses the unscented transformation is developed.

KURZFASSUNG

Die digitale Kommunikation ist in ihren verschiedenen Formen zu einem wesentlichen Teil des Alltags unserer Gesellschaft geworden. Dies hat die zunehmende Nachfrage nach digitalen Nachrichtensystemen zur Folge. Mobilfunknetze haben derzeit insgesamt 7,3 Milliarden Abonennten weltweit, wovon 1,4 Milliarden zur aktuellsten Mobilfunktechnologie der vierten Generation (4G) gehören. Bis 2022 soll die Gesamtmenge der Abonennten auf etwa 8,9 Milliarden und für 4G auf 4,3 Milliarden anwachsen. Ferner werden Anwendungen, die einen hohen Datendurchsatz benötigen, wie z.B. virtuelle Realitätssoftware, vorausgesehen. Die Kommunikationssysteme sollten die ebenfalls ansteigende Nachfrage nach machine to machine communications abdecken. Darunter sind Internet der Dinge (IoT) und vehicular ad hoc networks (VANETs) ebenso einbezogen wie die Fahrzeug-zu-Fahrzeug (V2V) und Fahrzeug-zu-Infrastruktur (V2I) Kommunikation. Um eine derartige Nachfrage unterstützen zu können, wird ein Wachstum der Datenübertragungsrate in der Größenordnung von 100 derzeit als Anforderung für die Standards der fünfte Generation (5G), die bereits 2020 im Einsatz sein sollten, betrachtet.

Um eine bessere Ausnutzung des knappen Spektrums zu ermöglichen, bietet sich der Einbau von Mehrantennensystemen in Kommunikationsgeräte als Schlüsseltechnologie an. Die vorliegende Arbeit legt den Schwerpunkt besonders auf Beamforming-Techniken, die das Strahlungsmuster des in Mehrantennensystemen angeordneten Geräte virtuell anpassen können, um das Signal aus einer erwünschten Richtung zu verstärken beziehungsweise die Störung aus weiteren Winkeln zu neutralisieren. Beamforming ermöglicht somit die räumliche Trennung von mehreren Quellen, die dasselbe Spektrum teilen, und kann ebenso verwendet werden, um elektronische Störer und Interferenzquellen abzuschwächen.

In der vorliegenden Arbeit wurden Beamforming-Verfahren und Frameworks entwickelt, um unterschiedliche Fälle des farbigen Rauschen, zweidimensionalen Mehrantennensystemen (URA) und breitbandigen Signalen zu untersuchen. Für Systemen mit farbigen Rauschen wurden Prewhitening, Rank Reduction und eine Transformation verwendet. Für URA und breitbandige Systemen wurde die Tensordarstellung benutzt und dabei ist die Tensorzerlegung anhand der Parallel Factor Analysis (PARAFAC) zusammen mit frequency invariant

beamformers (FIBs) verwendet worden. Zuletzt wurde mittels der Unscented Transformation mit geringem Rechenaufwand eine Evaluationsmethode entwickelt.

RESUMO

Comunicações digitais, seja por voz, mensagens instantâneas, vídeo ou outro conteúdo de mídia, se tornaram uma parte essencial da sociedade moderna. Como consequência, a demanda por sistemas avançados de comunicação está aumentando. Atualmente, as redes de comunicação móvel têm um total de 7,3 bilhões de assinaturas em todo o mundo, das quais 1,4 bilhão pertencem à rede mais nova da quarta geração (4G). Para 2022, são esperadas 8,9 bilhões de assinaturas, sendo 4,3 bilhões de assinaturas 4G. Além disso, também são previstas aplicações que requerem uma alta taxa de transmissão como a realidade virtual (VR). Os sistemas de comunicação também devem alocar a crescente demanda de comunicações máquina a máquina, incluindo a internet das coisas (IoT), redes veiculares e ad hoc (VANETs), como as redes veículo a veículo (V2V) e comunicações veículo a infraestrutura (V2I). Para suportar esta demanda, um aumento na escala de 100 vezes está sendo considerado como um requisito para os padrões de comunicação da futura quinta geração (5G), à qual tem seu emprego previsto para 2020.

Uma das tecnologias chaves para permitir uma melhor exploração do escarço espectro é a incorporação de arranjos de antenas aos dispositivos de comunicação. Particularmente, este trabalho tem foco em técnicas de formação de feixe, que podem virtualmente adaptar o padrão de irradiação de um arranjo de antenas de forma a amplificar sinais vindos de uma direção desejada e cancelar sinais vindos de outros ângulos. Portanto, formadores de feixe promovem a separação espacial de múltiplas fontes de sinal, que compartilham uma mesma banda do espectro e, além disso, também podem ser aplicados para mitigação de interferência.

Neste trabalho, técnicas e frameworks de formadores de feixe são desenvolvidos levando em conta ruído colorido, arranjos uniformes retangulares (URAs) e sinais banda larga. Para cenários com ruídos coloridos, técnicas de branqueamento, redução de posto e uma transformação são utilizadas. Para a URA e casos banda larga é utilizada uma notação tensorial e a decomposição por meio da análise de fatores paralelos (PARAFAC) é aplicada juntamente com formadores de feixe invariantes em frequência (FIBs). Por fim, um método de avaliação com baixo custo computacional é desenvolvida por meio da transformada da incerteza (UT).

Contents

Contents	11
Chapter I Introduction	15
1.1 Beamforming	16
1.2 Low Complexity Array Evaluation	19
1.3 Objectives	19
1.4 Overview and Contributions	20
1.5 Notation	21
Chapter II Beamforming in Colored Noise Scenarios	23
2.1 Data Model	24
2.2 State-of-the-art beamformers	25
2.2.1 Batch GSC	25
2.2.2 Least Mean Squares GSC (LMS-GSC)	27
2.2.3 Reduced Rank Least Mean Squares GSC (RR-LMS-GSC)	28
2.3 Proposed Solutions	29
2.3.1 High Accuracy Stochastically Prewhitened LMS-GSC (HASP-LMS-GSC)	29
2.3.2 Vandermonde Invariance Transformation (VIT) based HASP-LMS-GSC	31
2.3.3 Deterministic Prewhitening (DP)	32
2.3.4 Block-wise Reduced Rank Stochastic Gradient GSC (BW-RR-SG-GSC)	34
2.3.5 Computational Complexity Analysis	35
2.4 Simulations and Results	35
2.5 Summary	42
Chapter III Multidimensional Beamformer for Separable Arrays	43
3.1 Data Model	44
3.2 Batch GSC for multiple dimensions	45
3.3 Proposed R -Dimensional GSC	47
3.4 Diagonal Loading	49
3.5 Simulations and Results	50

3.6	Summary	53
Chapter IV Broadband Beamforming via Frequency Invariance Transformation and PARAFAC Decomposition Tracking		55
4.1	Broadband Signal Model	56
4.2	Multi-Dimensional Inverse Fourier Transform Frequency Invariant Beamformer (FIB) Design	60
4.3	Bank of FIBs as a Virtual Array of Instantaneous Mixtures	62
4.4	Tensor Decomposition for Steering Matrix Estimation	66
4.4.1	A Review on the PARAFAC Tensor Decomposition	66
4.4.2	Alternate Least Squares for Steering Matrix Estimation	67
4.5	Proposed Adaptive Framework for Broadband Signals	69
4.6	Simulations and Results	74
4.7	Summary	80
Chapter V Beamformer Performance Assessment via the Unscented Transformation		81
5.1	Data Model	83
5.2	Unscented Transformation	84
5.2.1	Univariate UT	84
5.2.2	Multivariate UT	86
5.3	Proposed Performance Assessment of Array Response via UT	88
5.3.1	Univariate case: Evaluating the SINR under DOA estimation error	88
5.3.2	Multivariate case: Evaluating the SINR under antenna element positioning error	90
5.4	Simulation Results	92
5.4.1	Performance Evaluation of the Univariate UT	94
5.4.2	Performance Evaluation of the Multivariate UT	95
5.5	Summary	98
Chapter VI Conclusions		99
6.1	General Conclusions	99
6.2	Recommendations for Future Research	100
Appendix A Blocking Matrix Computation		101
Appendix B Derivation of the Block-wise Stochastic Gradient GSC		105
List of Tables		109

List of Figures	111
Symbols	115
Acronyms	117
Bibliography	119
Erklärung	129

I

INTRODUCTION

Radio communications play a key role in modern society. For instance, it is estimated that in 2016, 7.3 billion mobile users were connected to the network and from these users, 6.3 billion are smartphone subscriptions. From these subscriptions, 1.5 billion belong to the latest long term evolution (LTE) or fourth generation (4G) technology [12]. Even though these devices are already practically ubiquitous, the demand for more data rate, better navigation accuracy and reliability will still grow at an estimated pace of 5 % to 10 % a year and it is expected to reach the mark of 8.9 billion subscriptions by 2022, from which approximately half billion should already be fifth generation (5G) subscribers [12].

This demand is expected to increase 100 fold by means of data rate per radio link in the 5G network. One of the reasons for the need of this growth is the addition of devices such as cars, machines and consumer electronics to the network, forming the so called Internet of things (IoT), and the insertion of new media content in form of virtual reality (VR) [71]. In order to increase data rate, antenna arrays are a natural evolution of communication systems, being adopted in a small scale of up to 8 antennas in current LTE cellular networks [33] and a larger number is expected for 5G systems [55].

Still according to [71], the currently used mobile spectrum, approximately between 800 MHz and 4 GHz, does not suffice the band needs for next generation wireless communication systems. This spectrum can only provide channels with bands not wider than 20 MHz. In order to increase bandwidth, the 5G next generation standard is suggesting the use of high frequency spectrum above 20 GHz. At this frequency range, the wavelengths are very small, only of a few millimeters and, thus, is the antenna size. The reduction of size reduces the amount of power received by the antenna, and, in addition, pathloss is greater at this frequency region. In order to compensate for the antenna size reduction and pathloss, antenna arrays can be used. The envisioned next generation mobile network is expected to start its deployment in 2020 and the usage of tens or even hundreds of antennas is under discussion [56, 75]. When multiple antennas are present on both communication ends, the

system is often called a multiple input multiple output (MIMO) system. When the number of antennas is high, the system is called a massive MIMO system. As the number of antennas is increased towards a massive MIMO application, the number of simultaneous transmissions to various users also increases [55].

A key technology in antenna arrays is beamforming. In essence, beamformers perform delay and sum operations in order to create constructive interference towards a desired signal direction [82]. The constructive interference towards a desired signal is the so called beam. In addition, extra information about interference position or statistical information extracted from received signals can be used to construct destructive interference towards the non desired signal direction. The usage of beamformers in practical applications still have some challenges such as cases where the signals are corrupted by noise, better ways to deal with big arrays, or even broadband signals. Each one of these challenges are depicted in Section 1.1.

This chapter is divided into five sections. In Section 1.1 makes an introduction about beamformers, their challenges and how they are exploited in this work. Section 1.2 describes the importance of an antenna array system evaluation and how it is approached in this dissertation, while the objectives of the work are depicted in Section 1.3. Section 1.4 describes how this work is divided. Section 1.5 shows the used mathematical notation.

1.1 Beamforming

Beamformers work from the simple idea of summing up similar waveforms from different antennas that in principle differ only by a delay. The delays can be electronically or digitally adjusted creating constructive and destructive interference in order to increase the desired signal's quality. Figure 1.1 shows beams formed from a constructive interference of 6 antennas.

The simplest is the delay and sum (DS) beamformer [82] which, as its name suggests, properly delays the signal acquired by each antenna and sums them up in order to form a constructive interference in the direction of the desired signal. More sophisticated beamforming solutions are found in the form of the linearly constrained minimum variance (LCMV), also known as the Capon beamformer [19] and the generalized sidelobe canceller (GSC) beamformers [40]. The last two beamformers require the estimation of the correlation matrices of received signal in order to compute the beamforming filters. In the system design these beamformers regard the noise as uncorrelated white noise, where in practice, they can also be correlated [25], causing their degradation. This work tackles colored noise scenarios for beamforming by using prewhitening techniques. On the other hand, prewhitening techniques also increase the systems computational complexity. Therefore a block wise

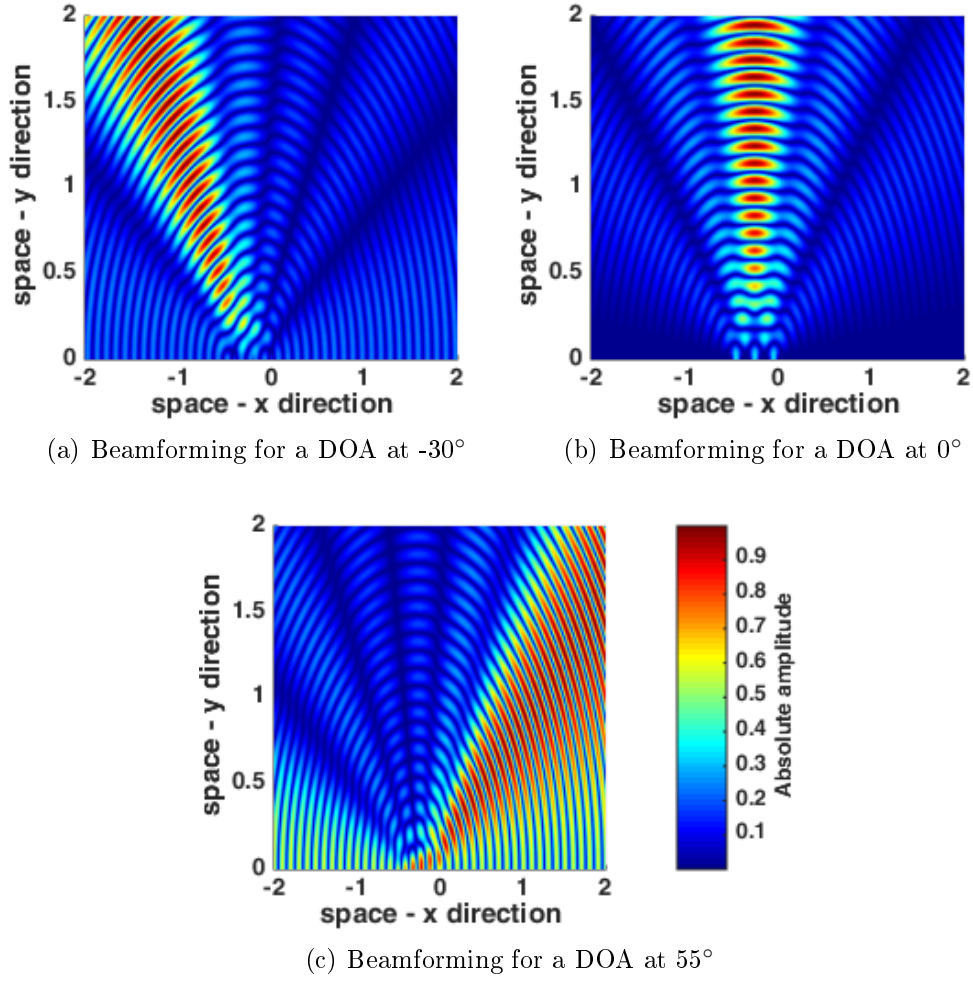


Figure 1.1: Senuoidal waves were generated at 5 Hz and virtually steered, for the sake of visualization of the beamforming. The constructive patterns are clearly visible and appear in dark red.

GSC algorithm is developed to allow for more flexibility and control of the final systems complexity by choosing a block size and number of iterations.

This work shows that when the array has a 2-dimensional separable structure such as an uniform rectangular array (URA) [73], beamformers can also be computed separately. When the structure is separated, the each column of the array can be regarded as snapshot containing information on the y -direction and each row as a snapshot containing samples of the x -direction. This virtually increases the number of snapshots at the price of resolution that may be desirable in massive MIMO scenarios [75]. This work also shows that besides increasing the number of snapshots and, therefore, enhancing correlation matrix estimation, the use of two smaller matrices instead of a bigger one can also reduce the computational time associated with beamforming algorithms.

For most beamformers in communications, the narrowband assumption is used for the

beamformer design. In cases where the narrowband scenario no longer holds, the signal band can be separated into shorter band ones so that each smaller band can be considered as narrow [64]. To avoid the increased complexity, extra antennas, such as in the of massive MIMO schemes, can be used to create frequency invariant beamformers (FIBs) [76]. The output of several independent FIBs contains instantaneous mixtures, similar to those found in narrowband scenarios. Therefore, these outputs can serve as input to narrowband algorithms which have rich literature [50].

Previous works using a bank of FIBs have tackled the narrowband beamforming step using independent component analysis (ICA) algorithms. Traditionally, ICA is based on the measure of the Gaussianity computed via Kurtosis, and, due to the central limit theorem, the greater is the Kurtosis, the more separated are the output signals [50]. Since the Gaussianity is used to separate the signals, then the standard ICA cannot be applied for Gaussian distributed signals.

One way to avoid the Gaussianity measurement problem is to use methods based on the non-whiteness of the signal. When a signal is uncorrelated, i.e. there is no time dependency between its samples, the signal is said to be white, since its spectrum should have about the same power for all frequencies. When there is a time correlation, the signal is, on the other hand, said to be non-white. In this work, the Gaussianity measure problem is avoided by the use of an algorithm that exploits non-whiteness [57].

In [57] the non-whiteness is exploited firstly by stacking time correlation matrices one on the side of the other in order to form a three dimensional representation, the so called tensor, as illustrate in Figure 1.2. Secondly, this tensor is decomposed via the parallel factors analysis (PARAFAC) multilinear decomposition [46], also known as the canonical decomposition (CANDECOMP) [21]. This decomposition results in factor matrices that contain estimators for the desired beamforming weights.

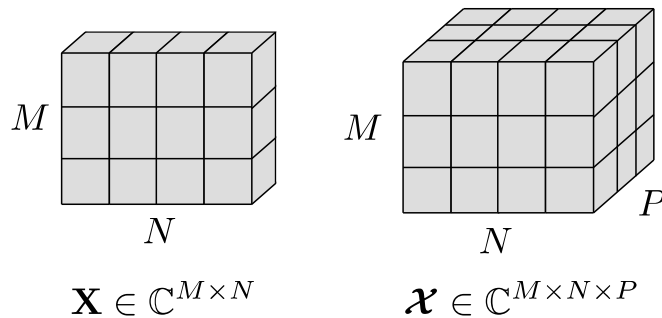


Figure 1.2: Illustration of a matrix of size $M \times N$ and 3-dimensional tensor of size $M \times N \times P$

In this work, in order to reduce its the complexity of [57] to online implementation levels, an adaptive approach is used [70]. However, the approach in [70] was originally proposed to

signals that have its statistical characteristics changing over time, i.e. non-stationary. Hence, this work proposes modification for the adaptive PARAFAC algorithms to be incorporated to beamforming by exploiting the non-whiteness property.

1.2 Low Complexity Array Evaluation

The beamforming techniques should be evaluated for different scenarios. For instance, scenarios with colored noise and positioning errors of the array assembly should be taken into account. With that in mind, it is desirable to study the impact that imperfections from non ideal equipment will cause to a beamformer. This task, however, may require a lot of computational power. One way to overcome this complexity is to make use of numerical analysis to simplify the computations.

One of the important imperfections that arises from a practical implementation of beamformers is the estimation of the direction of arrival (DOA). In the development of beamformers, engineers usually consider the DOA as known, however, in practice, the DOA estimation is not perfect and has errors associated to it. In this work, the DOA estimation impact on the SINR is evaluated by usage of the unscented transformation (UT) numerical tool for very low computational complexity when compared to very popular Monte Carlo simulations with good results [8].

Another problem from array implementation is the placement of the array elements that also brings along positioning deviations. Since the positioning errors occur for all the elements, for this computation, an amount of variables equal to the number of array elements has to be considered, creating a multi-variate problem for the UT. Even though the multi-variate complexity is increased, this work shows that depending on the deviation caused by these imperfections and the number of antennas, the low complexity computation using the UT is still advantageous.

1.3 Objectives

Next generation communication systems are predicted to have an increased number of antenna elements. The broad objective of this work is to increase current wireless systems data rates via beamforming. Beamforming allows for better signal to noise ratio and interference mitigation.

The implementation of beamformers in real scenarios has some challenges. One of them is to maintain proper function under colored noise. In colored noise scenarios, the noise from each antenna is correlated to the other, a situation that should be encountered in practical arrays. Therefore, the first specific objective of this work is to develop beam-

forming techniques that work under colored noise scenarios. In this work, the usage of two prewhitening techniques to mitigate the colored characteristics of the noise is proposed. Since the prewhitening step increases the complexity, a block-wise processing beamforming technique is proposed to increase the system design flexibility in terms of complexity.

When the number of antennas increases, the computational complexity also grows. Therefore, the second specific objective of this work is to reduce computational complexity when a high number of antennas is available, yet, maintaining a good interference mitigation. For that, two schemes are proposed. The first approach computes smaller correlation matrices for each dimension and each corresponding beamformer is computed. Then, the beamformers for each dimension are combined. The second scheme considers a broadband scenario and converts it into a narrowband one by using a bank of FIBs. The output of this bank of FIBs is used to create a correlation tensor which can be decomposed. Such decomposition generates a demixing matrix that is used to form a final broadband beamformer.

Finally, beamformers should be robust against estimation errors of the source direction of arrival (DOA) and imperfections on the placement of the array elements. The performance assessment of beamformers for different scenarios can be time demanding. Therefore, the final specific objective is to reduce the complexity of the evaluation of beamforming in the presence of DOA estimation error and positioning estimation error. In this work, the evaluation using the UT numerical tool is proposed.

1.4 Overview and Contributions

This thesis is divided into five chapters including this introduction. Chapter II contains a description of antenna arrays in colored noise scenarios and how to reduce its effects on adaptive beamforming. Moreover, reduced complexity is also achieved after deriving a block-wise adaptation algorithm. The main contributions found in Chapter II are the usage of deterministic and stochastic prewhitening prior to the beamformer, the usage of the Vandermonde invariant transformation (VIT) to increase beamforming accuracy and the derivation of a block-wise GSC for computational complexity reduction.

Chapter III describes how the data received from separable arrays can be modeled into a multidimensional form using the tensor notation. Moreover, with the insights eased by the use of this notation, the multidimensional structure is used to virtually increase the amount of samples improving beamforming filter quality and reduce computational complexity. In Chapter III, the contribution is the adoption of the concept of array separability allowing for the application of multidimensional operations to virtually increase the number of samples. Also, the multidimensional operations generate a group of smaller correlation matrices instead of a big one, which in turn, reduce computational complexity of batch algorithms.

Chapter IV reviews the concept of FIBs and its applications as a bank of FIBs. The PARAFAC decomposition is also reviewed and it is applied with the bank of FIBs concept. The contribution in Chapter IV is creation of a framework that applies a bank of FIBs with the PARAFAC decomposition. The framework also includes a novel way to create the adaptive tensor and an adaptive filter estimation step is added to the adaptive PARAFAC tracking algorithm for faster convergence and improved accuracy.

Chapter V evaluates the impact of imperfections on beamformers. In Chapter V, the effects of these imperfections are modeled in a novel manner by using the unscented transformation (UT). The proposed UT based evaluation assessment shows a considerable reduction in computational time for antenna array systems.

Finally, Chapter VI draws the conclusions about this work.

1.5 Notation

Along this work, the following notation is used. Scalars are denoted by lower-case letters (a, b, \dots), vectors are written as boldface lower-case letters ($\mathbf{a}, \mathbf{b}, \dots$), matrices as boldface capitals ($\mathbf{A}, \mathbf{B}, \dots$), and tensors as boldface calligraphic letters ($\mathcal{A}, \mathcal{B}, \dots$). The superscripts T , H , and $*$ represent transpose, Hermitian transpose and complex conjugate of a matrix, respectively. The notation $\mathbf{A}(:, i) \in \mathbb{C}^{R \times 1}$ represents a column vector denoting the i -th column of $\mathbf{A} \in \mathbb{C}^{R \times I}$. The operator $\text{vec}(\mathbf{A})$ results in a vector by concatenating the columns of the matrix \mathbf{A} one on top of the other. The notation $[\mathcal{T}]_{(r)}$ is the r -mode matrix unfolding of \mathcal{T} and $\mathcal{T} \times_r \mathbf{A}$ is the r -th mode product between the tensor \mathcal{T} and the matrix \mathbf{A} . Moreover, the Kronecker product and outer product operators are denoted by \otimes and \circ , respectively. The Khatri-Rao operator denoted as \diamond is defined as the column-wise Kronecker product. The operator $\text{E}\{\cdot\}$ stands for the expected value operation.

II

BEAMFORMING IN COLORED NOISE SCENARIOS

Signal processing is an important field in many areas such as satellite communications [5, 7], navigation [77], speech [3, 10, 11, 78], RADAR [87] and many communication systems including antenna array communications [6, 47, 64], in which beamforming plays an important role [48, 69, 80, 87]. In the literature, there are several adaptations of Direction of Arrival (DOA) estimation schemes for colored noise scenarios [25, 28, 44], since the DOA information is crucial for the beamformer. The addition of such constraints led to the development of beamformers such as the Direct Form Processor (DFP), which includes the Linearly Constrained Minimum Variance (LCMV) and Linearly Constrained Constant Modulus (LCCM) [84], and the Generalized Sidelobe Canceler (GSC) [47]. For real time applications, the necessity for adaptive algorithms grows and, with this need, adaptive versions of the GSC were proposed in earlier works [23, 84, 85].

However, when the number of elements in a sensor array is high in the sense that there are more antennas at the receiver than users, e.g. in future 5G networks, these algorithms suffer from computational complexity increase. Therefore, recently, adaptive reduced rank DFP and GSC schemes were also proposed in order to reduce the dimensionality of the adaptive filters. The rank reduction step also has a noise removal effect, thus showing an enhanced performance [84, 85]. These works use the constant modulus (CM) cost function [23] due to its improved accuracy for constant envelope signals. Yet, adaptive beamforming techniques using the GSC usually assume uncorrelated white noise in the receivers, which is not realistic since noise is usually colored, i.e. spatially correlated [25].

For colored noise scenarios, prewhitening schemes have been successfully applied in combination with DOA estimation [28, 44] and audio signal processing schemes [45]. The prewhitening schemes are divided into stochastic [44, 45] and deterministic prewhitening [28]. In deterministic prewhitening, the noise may have a specific structure which can be exploited,

while in the stochastic prewhitening, no structure is assumed. Moreover, there are also multidimensional prewhitening schemes for the case that the data has a tensor structure [26].

In this work, the least mean squares GSC (LMS-GSC) and the Reduced Rank LMS-GSC (RR-LMS-GSC) are extended for colored noise scenarios by incorporating a prewhitening step. The prewhitened GSC schemes considering the deterministic prewhitening [28] and the stochastic prewhitening [44, 45] are proposed. The colored noise is usually concentrated in certain direction. Therefore, to further enhance the GSC, the Vandermonde Invariance Transformation (VIT) [51] is also applied as a preprocessing step. The colored noise can be also concentrated close to the desired signal direction, therefore a prewhitening step is also still needed along the VIT. The stochastic prewhitening needs the computation of one SVD at each iteration. In order to reduce the complexity of the stochastic prewhitening, the block-wise reduced rank stochastic gradient GSC (BW-RR-GSC) beamformer is proposed.

This chapter is divided into four sections including this introduction. Section 2.1 describes the used data model. In Section 2.2 the state-of-the-art beamformers are reviewed. In Section 2.3 a high accuracy and low complexity GSC schemes are proposed by incorporating prewhitening steps, the VIT and a block-wise modification for colored noise scenarios. In Section 2.4, simulations are shown and the results are drawn. Finally, Section 2.5 makes the conclusions.

2.1 Data Model

We assume that d sources are transmitting different symbols at the n -th time instant. Since the sources are far away from the receiver, the narrowband wave fronts are considered planar. A uniform linear array (ULA) is assumed and contains M isotropic sensor elements with an inter-element spacing of Δ wavelengths. Therefore, the received symbols are mathematically represented as

$$\mathbf{x}(t) = \mathbf{a}(\theta_0)s(t) + \mathbf{A}_{\text{int}}(\boldsymbol{\theta}_{\text{int}})\mathbf{s}_{\text{int}}(t) + \mathbf{n}^{(c)}(t), \quad (2.1)$$

where $\mathbf{x}(t) = [x_0(t), \dots, x_{M-1}(t)]^T$ is the vector containing the received symbols at time instant t , $s(t)$ is the desired signal, $\mathbf{s}_{\text{int}}(t)$ is a vector with the interference symbols from the $d - 1$ interferes and $\mathbf{n}^{(c)}(t)$ contains colored noise samples at the sensor elements. Note that $\mathbf{n}^{(c)}(t) = \mathbf{L}\mathbf{n}(t)$, where $\mathbf{n}(t)$ contains i.i.d. noise samples with circularly symmetric complex Gaussian distributions. The matrix $\mathbf{L} \in \mathbb{C}^{M \times M}$ stands for the correlation factor matrix. For the special case where \mathbf{L} is the identity matrix the noise becomes white at the sensors. The vector $\mathbf{a}(\theta_0)$ is the steering vector with a Vandermonde structure

$$\mathbf{a}(\theta_0) = [1, e^{j\phi_0}, e^{j2\phi_0}, \dots, e^{j(M-1)\phi_0}]^T, \quad (2.2)$$

where θ_0 is the azimuth of the DOA and $j = \sqrt{-1}$. The matrix $\mathbf{A}_{\text{int}}(\boldsymbol{\theta}_{\text{int}})$ is the steering matrix containing all the steering vectors of the interfering signals

$$\mathbf{A}_{\text{int}}(\boldsymbol{\theta}_{\text{int}}) = \begin{bmatrix} 1 & 1 & \dots & 1 \\ e^{j\phi_1} & e^{j\phi_2} & \dots & e^{j\phi_{d-1}} \\ e^{j2\phi_1} & e^{j2\phi_2} & \dots & e^{j2\phi_{d-1}} \\ \vdots & \vdots & \ddots & \vdots \\ e^{j(M-1)\phi_1} & e^{j(M-1)\phi_2} & \dots & e^{j(M-1)\phi_{d-1}} \end{bmatrix} \in \mathbb{C}^{M \times d-1}, \quad (2.3)$$

where their corresponding DOAs are comprised in the vector $\boldsymbol{\theta}_{\text{int}} \in \mathbb{C}^{d-1 \times 1}$ and the DOAs are represented by the spatial frequencies, i.e. spatially related phase delays, $\phi_0 = -2\pi\Delta \sin \theta_0$ and $\boldsymbol{\phi}_{\text{int}} = -2\pi\Delta \sin \boldsymbol{\theta}_{\text{int}} \in \mathbb{C}^{d-1 \times 1}$.

More generally, assuming a sliding window in which at time t a snapshot of the current and the $N - 1$ previously transmitted symbols are allocated into a vector \mathbf{s} and collecting the interfering signals into a matrix $\mathbf{S}_{\text{int}}(t) \in \mathbb{C}^{N \times d-1}$, the model is rewritten in a compact format

$$\mathbf{X}(t) = \mathbf{a}(\theta_0)\mathbf{s}^T(t) + \mathbf{A}_{\text{int}}(\boldsymbol{\theta}_{\text{int}})\mathbf{S}_{\text{int}}^T(t) + \mathbf{N}^{(c)} \in \mathbb{C}^{M \times N}, \quad (2.4)$$

where $\mathbf{X}(t) = [\mathbf{x}(t - N + 1), \dots, \mathbf{x}(t)]$ and $\mathbf{N}^{(c)} = \mathbf{L} \cdot \mathbf{N} \in \mathbb{C}^{M \times N}$. The matrix $\mathbf{N} \in \mathbb{C}^{M \times N}$ contains the N white noise samples for all M sensors in the same manner as \mathbf{X} contains N signal plus noise samples from the M sensors. The variable $\mathbf{s}(t) \in \mathbb{C}^{N \times 1}$ has the N latest samples for the desired signal and $\mathbf{S}_{\text{int}}(t) \in \mathbb{C}^{N \times d-1}$ has the N latest samples for the $d - 1$ interfering sources.

Here, the received symbols $\mathbf{X}(t)$ and the DOA of the desired signal θ_0 are assumed as known by the receiver and it is desired to find $\hat{s}(t)$, which is an estimate of $s(t)$. To find the DOA, the reader is referred to [1, 25, 28, 44] or, alternatively, it is assumed that the position of the transmitter with respect to the receiver is known.

2.2 State-of-the-art beamformers

2.2.1 Batch GSC

The GSC algorithm turns a constrained problem into an unconstrained one by introducing a blocking matrix, which is the orthogonal complement of the constraint $\mathbf{a}(\theta_0)$. In this case, the constraint is formed based on the steering vector of the desired signal, formed with the DOA estimate.

The classical GSC [40] block diagram is depicted in Figure 2.1.

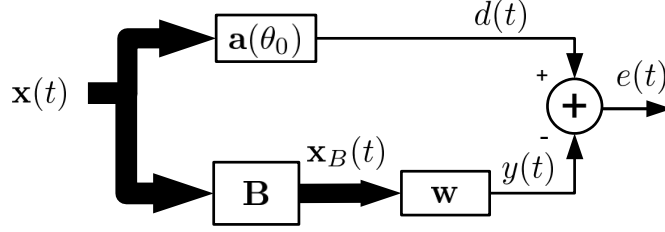


Figure 2.1: Batch GSC block diagram

First, in the upper part of Figure 2.1, the signal $d(t) = \mathbf{a}^H(\theta_0)\mathbf{x}(t)$ is extracted by a delay and sum filter $\mathbf{a}(\theta_0)$ with taps computed in the same manner as (2.3) and steered in the direction of the signal $s(t)$. This signal is considered the desired signal, thus the notation $d(t)$, and the remaining $d - 1$ signals are the interference signals in accordance to Section 2.1.

In the bottom part of Figure 2.1, a blocking matrix $\mathbf{B} \in \mathbb{C}^{(M-1) \times M}$, such that $\mathbf{B} \cdot \mathbf{a}(\theta_0) = \mathbf{0}$, is used to extract the interference signals $\mathbf{x}_B(t) = \mathbf{B} \cdot \mathbf{x}(t)$. Consequently, \mathbf{B} blocks the desired signal and lets ideally only $\mathbf{A}_{\text{int}}(\theta_{\text{int}})\mathbf{s}_{\text{int}}(t)$ pass. Details on the computation of the blocking matrix \mathbf{B} are shown in Appendix 6.2. In an ideal system, the filter \mathbf{w} is computed as the Wiener solution of $|d(t) - \mathbf{w}^H \mathbf{x}_B(t)|^2$ to give an interference-removing, or sidelobe canceller, filter:

$$\mathbf{w} = \mathbf{R}_{\mathbf{x}_B}^{-1} \mathbf{r}_{\mathbf{x}_B d}, \quad (2.5)$$

where, $\mathbf{R}_{\mathbf{x}_B}$ and $\mathbf{r}_{\mathbf{x}_B d}$ are the correlation matrix of $\mathbf{x}_B(t)$ and the cross correlation vector between $\mathbf{x}_B(t)$ and $d(t)$, respectively. The total filter is

$$\mathbf{w}_{\text{gsc}} = \mathbf{a}(\theta_0) - \mathbf{B}^H \mathbf{w}, \quad (2.6)$$

thus the left-hand part of Figure 2.1 corresponding to $\mathbf{a}(\theta_0)$ and \mathbf{B} are the static part of the system and the right-hand part corresponding to \mathbf{w} is adaptive.

The batch GSC follows the same principle of 2.5, but replaces the correlation and cross correlation with their estimates using N samples:

$$\hat{\mathbf{R}}_{\mathbf{x}_B}^{-1} = \frac{1}{N} \mathbf{B} \mathbf{X}(t) \mathbf{X}^H(t) \mathbf{B}^H, \quad (2.7)$$

$$\hat{\mathbf{r}}_{\mathbf{x}_B d} = \frac{1}{N} \mathbf{B} \mathbf{X}(t) \mathbf{d}(t)^*, \quad (2.8)$$

where $\mathbf{d}(t) = [d(t - N + 1), \dots, d(t)]^T$. The batch algorithm can be either adapted snapshot by snapshot via a sliding window or in a block-wise manner by storing snapshots.

2.2.2 Least Mean Squares GSC (LMS-GSC)

The Least Mean Squares GSC (LMS-GSC) is similar to the batch GSC except that the correlation and cross correlation are roughly estimated with a single snapshot. In Fig. 2.2, the input signal $\mathbf{x}(t)$ passes through a beam pointed at the desired signal direction θ_0 generating $d(n) = \mathbf{a}^H(\theta_0)\mathbf{x}(t)$. The same input signal also passes through a blocking matrix \mathbf{B} which is the orthogonal complement of the constraint $\mathbf{a}(\theta_0)$. The filter \mathbf{w} should then be adjusted so that it generates the interference signal $y(t)$ that is subtracted from the desired signal $d(t)$.

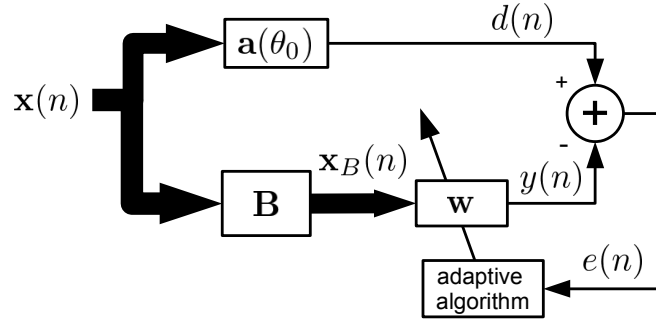


Figure 2.2: LMS-GSC block diagram

In Fig. 2.2, $y(t)$ is given by

$$y(n) = \mathbf{w}^H \mathbf{x}_B(n), \quad (2.9)$$

where $\mathbf{x}_B(n) = \mathbf{B}\mathbf{x}(t)$. As shown in Fig. 2.2, the error signal $e(t)$ is used by the adaptive algorithm to adjust \mathbf{w} . Once \mathbf{w} converges, $\hat{s}(t) = e(t)$. Note that $e(t)$ is free from interference is also the system's output signal.

The adaptation of \mathbf{w} is computed via stochastic gradient of the following cost function

$$J_{\text{lms}}(\mathbf{w}) = E\{|d(t) - \mathbf{w}^H \mathbf{x}_B(t)|^2\} \quad (2.10)$$

which gives the update rule for the adaptive part

$$\mathbf{w}(t+1) = \mathbf{w}(t) + \mu_{\text{lms}} \nabla_{\mathbf{w}} J_{\text{lms}}(\mathbf{w}) \quad (2.11)$$

with μ_{lms} being the step size for the LMS-GSC.

We use the instantaneous estimates $\hat{\mathbf{R}}_{\mathbf{xx}} = \mathbf{x}(n)\mathbf{x}^H(n)$ and $\hat{\mathbf{r}}_{dx} = d(n)\mathbf{x}(n)$ [47] to find the stochastic gradient:

$$\hat{\nabla}_{\mathbf{w}} J_{\text{lms}} = 2\mathbf{B}\mathbf{x}(t)\mathbf{x}^H(t)\mathbf{B}^H\mathbf{w} - 2\mathbf{B}d(t)\mathbf{x}(t). \quad (2.12)$$

Now the stochastic gradient is inserted into LMS update rule for the GSC [47]:

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu_{\text{lms}} \mathbf{B} \mathbf{x}(n) \mathbf{x}^H(n) (\mathbf{a}(\theta_0) - \mathbf{B}^H \mathbf{w}). \quad (2.13)$$

2.2.3 Reduced Rank Least Mean Squares GSC (RR-LMS-GSC)

Regular LMS algorithms have been reported to have a slow convergence for high number of sensor elements M [85]. To overcome these drawbacks, a reduced rank (RR) adaptive algorithm for the GSC scheme was developed in [84, 85] and is depicted in Fig. 2.3. The

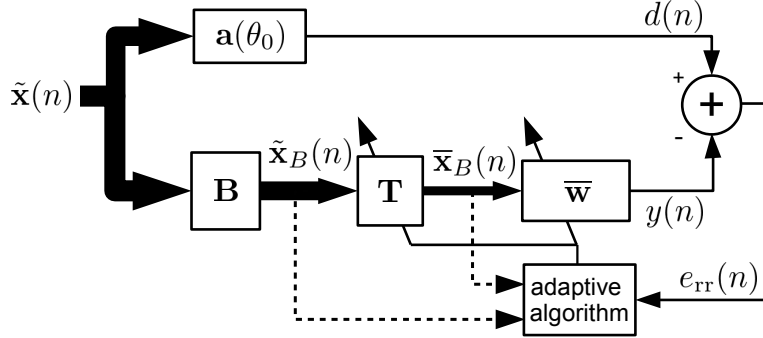


Figure 2.3: RR-LMS-GSC block diagram

process is similar to that described in Section 2.2.2, except that the blocked signal passes through a transformation matrix $\mathbf{T} \in \mathbb{C}^{(M-1) \times r}$ that performs a dimensionality reduction prior from being filtered by a reduced filter. As derived in [23], $\tilde{\mathbf{x}}(t) = \mathbf{e}_{\text{rr}}^*(t) \mathbf{x}(t)$ is used instead of $\mathbf{x}(t)$ for the input signal, where $e_{\text{rr}}(t) = \mathbf{w}^H(t) \mathbf{x}(t)$ is the output signal for the RR algorithm as indicated by the subscript and $\mathbf{w}(t) = \mathbf{a}(\theta_0) - \mathbf{B}^H \mathbf{T}(t) \bar{\mathbf{w}}(t)$. The CM error and its quadratic mean describes the CM cost function:

$$J_{\text{cm}}(\mathbf{w}) = E\{|\mathbf{w}^H \tilde{\mathbf{x}}(t) - \nu|^2\}, \quad (2.14)$$

where ν is constant set to 1 for unit variance signals. The filter \mathbf{w} can be represented in terms of time evolving $\mathbf{T}(t)$ and $\bar{\mathbf{w}}(t)$ for the adaptation process. Thus, the expanded cost function becomes

$$J_{\text{cm}}(\mathbf{T}(t), \bar{\mathbf{w}}(t)) = E\{|\mathbf{a}(\theta_0) - \mathbf{B}^H \mathbf{T}(t) \bar{\mathbf{w}}(t)|^H \tilde{\mathbf{x}}(t) - \nu|^2\}, \quad (2.15)$$

with $\bar{\mathbf{w}}(t)$ being the reduced rank filter of size r , where $d \leq r < M$. Therefore, it can be clearly seen that there is a trade off between the reduced rank size and the number of the degrees of freedom. As shown in [84, 85], the choice of r for values equal or slightly above d gives the best results, where the estimation of d can be performed by several methods [27, 29, 30]. The stochastic gradient of the previous cost function with respect to

$\mathbf{T}(t)$ and $\bar{\mathbf{w}}(t)$ gives their LMS update rules respectively

$$\mathbf{T}(t+1) = \mathbf{T}(t) + \mu_{\mathbf{T}} e_{\text{cm}}^*(t) \tilde{\mathbf{x}}_{\mathbf{B}}(t) \bar{\mathbf{w}}^H(t), \quad (2.16)$$

$$\bar{\mathbf{w}}(t+1) = \bar{\mathbf{w}}(t) + \mu_{\bar{\mathbf{w}}} e_{\text{cm}}^*(t) \bar{\mathbf{x}}_{\mathbf{B}}(t), \quad (2.17)$$

where $e_{\text{cm}} = 1 - \mathbf{w}(t)^H \tilde{\mathbf{x}}(t)$ is the CM error. Also note that a new variable is introduced in (2.17) which corresponds to the filter input signal $\bar{\mathbf{x}}_{\mathbf{B}}(t) = \mathbf{T}^H \tilde{\mathbf{x}}_{\mathbf{B}}(t)$, i.e. after the blocked signal transformation.

2.3 Proposed Solutions

In this section, different variations for the prewhitened adaptive GSC and prewhitened adaptive RR-GSC are proposed. The High Accuracy Stochastic Prewhitening (HASP) method that makes usage of one SVD per sample is shown in Section 2.3.1. The VIT is also used along the HASP to create the VIT-HASP solution. In the phase perspective, the VIT keeps the Vandermonde structure in a desired direction and changes the directions of the interferent sources. From the spatial power sense, it acts as beamformer conserving power at a direction of interest and reduces the power in other directions. The VIT results in a virtual array that is further used by the adaptive GSC. Since the direction where the noise power is concentrated is not known, it can be close to the desired signal making the prewhitening still required, thus forming the VIT-HASP. In Section 2.3.3, the Deterministic Prewhitening (DP) method is shown for the cases where the correlation structure of the noise is known. Since the prewhitening step requires one SVD per sample for sample-wise adaptive algorithms, a new block-wise reduced rank stochastic gradient GSC beamformer (BW-RR-SG-GSC), which requires only one SVD per block, is also presented.

2.3.1 High Accuracy Stochastically Prewhitened LMS-GSC (HASP-LMS-GSC)

We start with the traditional stochastic prewhitening method for estimation of the noise correlation factor matrix and reduction of the effects of colored noise as proposed by [45]. First, it is assumed that samples are free from signal components, i.e., only with noise, so that the noise sample correlation matrix can be given as

$$\hat{\mathbf{R}}_{\mathbf{NN}} = \frac{1}{N_{\text{sf}}} \mathbf{N}^{(\text{c})'} \cdot (\mathbf{N}^{(\text{c})'})^H, \quad (2.18)$$

where N_{sf} stands for the number of signal free samples and $\mathbf{N}^{(\text{c})'}$ is the colored noise snapshots matrix. Since $\hat{\mathbf{R}}_{\mathbf{NN}}$ is Hermitian and positive-definite the Cholesky decomposition can

be performed such that

$$\hat{\mathbf{R}}_{\mathbf{NN}} = \hat{\mathbf{L}} \cdot \hat{\mathbf{L}}^H, \quad (2.19)$$

with the estimated correlation factor matrix $\hat{\mathbf{L}}$ one can perform a prewhitening trying to revert the effects of the correlated noise in (2.1) by simply multiplying the data by the inverse of the estimated correlation factor matrix:

$$\mathbf{X}'(t) = \hat{\mathbf{L}}^{-1} \cdot \mathbf{X}(t), \quad (2.20)$$

where $\mathbf{X}'(t)$ is the prewhitened matrix of $\mathbf{X}(t)$. Once the number of signals d is known, an SVD low rank approximation on $\mathbf{X}'(t)$ can be applied, resulting to $\mathbf{X}^{[s]}(t)$. In order to estimate d , the reader is referred to model order selection schemes [27, 29, 30]. It is important to notice that this low rank approximation is not related to the rank reduction step performed by the adaptive algorithms. Finally, the noise and signal subspace are separated using the dewhitening step, which means multiplying it by the estimated correlation factor matrix $\hat{\mathbf{L}}$. This leads to the final input data:

$$\mathbf{X}^{[s]'}(t) = \hat{\mathbf{L}} \cdot \mathbf{X}^{[s]}(t). \quad (2.21)$$

In Fig. 2.4, (2.4) is used as input of the system for the HASP solution. Then, the SVD is applied in each iteration with the purpose of dewhitening. This results in a dewhitened data vector $\mathbf{x}^{[s]'}(t)$. The dewhitened data can now be inserted into the adaptive LMS-GSC or RR-LMS-GSC that will update the beamforming filter.

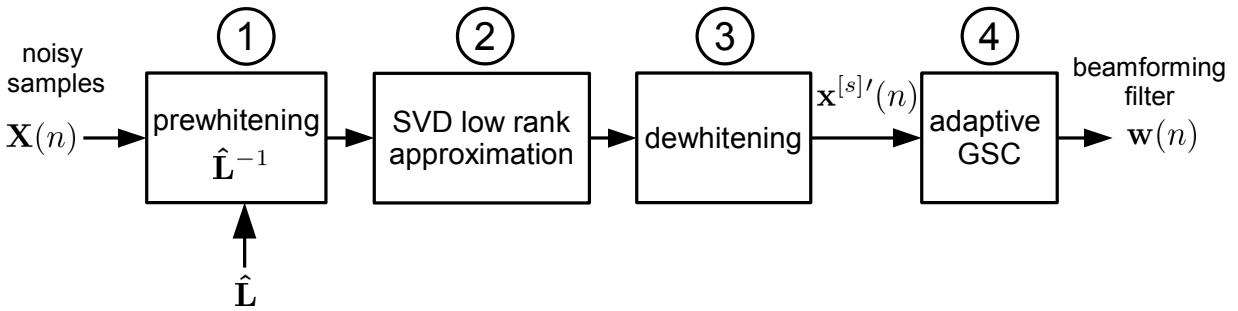


Figure 2.4: High accuracy stochastic prewhitening block diagram

If the algorithm is element-wise adaptive, then only the latest dewhitened snapshot $\mathbf{x}^{[s]'}(t)$ is used at the iteration keeping in mind that $\mathbf{X}^{[s]'}(t) = [\mathbf{x}^{[s]'}(t - N + 1), \dots, \mathbf{x}^{[s]'}(t)]$. This snapshot can be directly inserted into the update rule (2.11) and update rules (2.17) and (2.16). The block-wise adaptation in Section 2.3.4 uses all the N snapshots.

2.3.2 Vandermonde Invariance Transformation (VIT) based HASP-LMS-GSC

The purpose of the VIT is to transform the input signal in a way that the Vandermonde structure is preserved [51]. Since there are always imperfections on the array, the reader is referred to [66] for array interpolation in order to compensate such imperfections. First, the VIT matrix $\mathbf{T}_{\text{vit}} \in \mathbb{C}^{M \times M}$ is defined and it is multiplied with a vector that has a Vandermonde structure $\mathbf{v} = [1, e^{j\phi}, \dots, e^{j(M-1)\phi}]^T$. In [51] it is shown that

$$\tilde{\mathbf{v}} = \mathbf{T}_{\text{vit}} \cdot \mathbf{v} = \left(\frac{e^{j\phi} - \kappa}{1 - \kappa} \right)^{M-1} \cdot \begin{pmatrix} 1 \\ e^{j\nu} \\ \vdots \\ e^{j(M-1)\nu} \end{pmatrix}, \quad (2.22)$$

where κ is a complex parameter that originates a phase gain. From Eq. (2.22) it is inferred that the original phase ϕ is mapped into a new phase ν , but the Vandermonde structure is still present. More specifically, the new virtual phase is

$$\nu = \arctan \left(\frac{2K \cdot \sin \phi}{1 - K^2 + (1 + K^2) \cdot \cos \phi} \right), \quad (2.23)$$

where $K = (\kappa + 1)/(\kappa - 1)$. From (2.23) it is clear that the phase remains the same for $\phi = 0$. Relating these phases to the spatial frequencies ϕ_0 and ϕ_{int} , one shall shift the sources prior to applying the VIT by multiplying the input signal by $\mathbf{P}_{\phi_0} = \text{diag}([1, e^{-j\phi_0}, \dots, e^{-j(M-1)\phi_0}])$, so that the desired signal virtually impinges from 0° and no phase amplification is noticed. Therefore, the shift and transformation are applied to (2.1) leading to

$$\tilde{\mathbf{x}}(t) = \mathbf{T}_{\text{vit}} \mathbf{P}_{\phi_0} \mathbf{a}(\theta_0) s(t) + \mathbf{T}_{\text{vit}} \mathbf{P}_{\phi_0} \mathbf{A}_{\text{int}}(\boldsymbol{\theta}_{\text{int}}) \mathbf{s}_{\text{int}}(t) + \mathbf{T}_{\text{vit}} \mathbf{P}_{\phi_0} \mathbf{n}^{(c)}(t) \quad (2.24)$$

$$= \tilde{\mathbf{a}}(\theta_0) s(t) + \tilde{\mathbf{A}}_{\text{int}}(\boldsymbol{\theta}_{\text{int}}) \mathbf{s}_{\text{int}}(t) + \tilde{\mathbf{n}}^{(c)}(t). \quad (2.25)$$

The vector $\tilde{\mathbf{a}}(\theta_0)$ and the matrix $\tilde{\mathbf{A}}_{\text{int}}(\boldsymbol{\theta}_{\text{int}})$ still have a Vandermonde structure and $\tilde{\mathbf{n}}^{(c)}(t)$ is still a colored noise with its original color changed by the VIT so that $\tilde{\mathbf{n}}^{(c)}(t) = \mathbf{L}' \cdot \mathbf{n}(t)$ and $\mathbf{L}' = \mathbf{T}_{\text{vit}} \cdot \mathbf{P}_{\phi_0} \cdot \mathbf{L}$. As shown in [51], Equation (2.22) does not change the amplitude of \mathbf{v} at $\phi = 0$ but it is changed for $\phi \neq 0$. By setting $K < 1$ the amplitude away from 0° is diminished making the VIT to operate as a beamformer. Thus, the interference and noise not in the direction of the desired signal is reduced.

The noise can have most of its power close to the desired signal and thus only being partially removed by the VIT. Therefore, the usage of prewhitening is still needed. With the transformed noise correlation model, the HASP can now be directly applied using the

estimate

$$\hat{\mathbf{L}}' = \mathbf{T}_{\text{vit}} \cdot \mathbf{P}_{\phi_0} \cdot \hat{\mathbf{L}}. \quad (2.26)$$

Figure 2.5 summarizes the VIT-HASP process. First the noisy signal is shifted by \mathbf{P}_{ϕ_0} . At

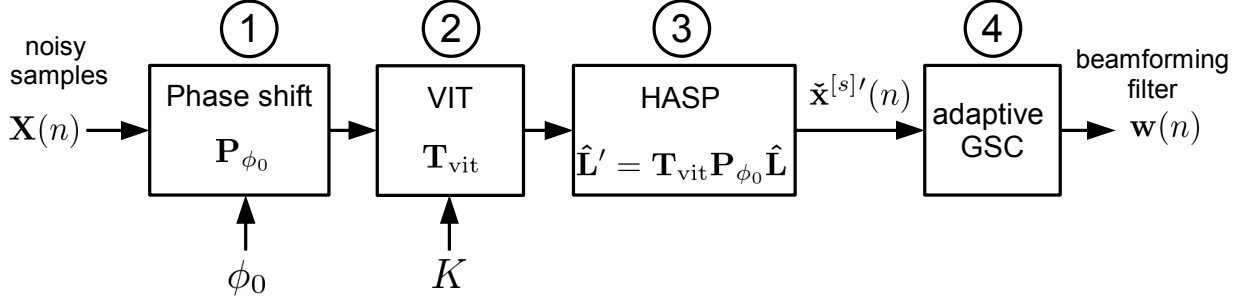


Figure 2.5: VIT-HASP block diagram

Step 2 the VIT is incorporated. After that, the signal is ready to pass through the modified HASP, where $\hat{\mathbf{L}}'$ is used in place of $\hat{\mathbf{L}}$. The result is a dewhitened data $\tilde{\mathbf{x}}^{[s]'}(n)$ to be used as input to the adaptive GSC algorithms. Details on the computation of \mathbf{T}_{vit} are found in [51].

2.3.3 Deterministic Prewhitening (DP)

In Subsections 2.3.1 and 2.3.2, the structure of the noise covariance matrix is unknown. However, for highly correlated environments and for known covariance structure, a deterministic prewhitening (DP) can be employed with the expense of one sensor as explained in [28]. In Fig. 2.6 signal-free samples are first inserted into the system in order to estimate the noise structure with its correlation level. With the noise structure and correlation level in step 2 of Fig. 2.6, the prewhitening matrix in step 3 of Fig. 2.6 can be built. In the last step the input data is multiplied by the generated matrix to produce the whitened data.

Here, it is assumed that the noise is correlated as proposed by [22]:

$$n_{m+1}^{(c)}(t) = \rho \cdot n_m(t) + \sqrt{1 - |\rho|^2} \cdot n_{m+1}(t), \quad (2.27)$$

where $0 \leq \rho < 1$ stands for the noise correlation coefficient. Once the noise correlation

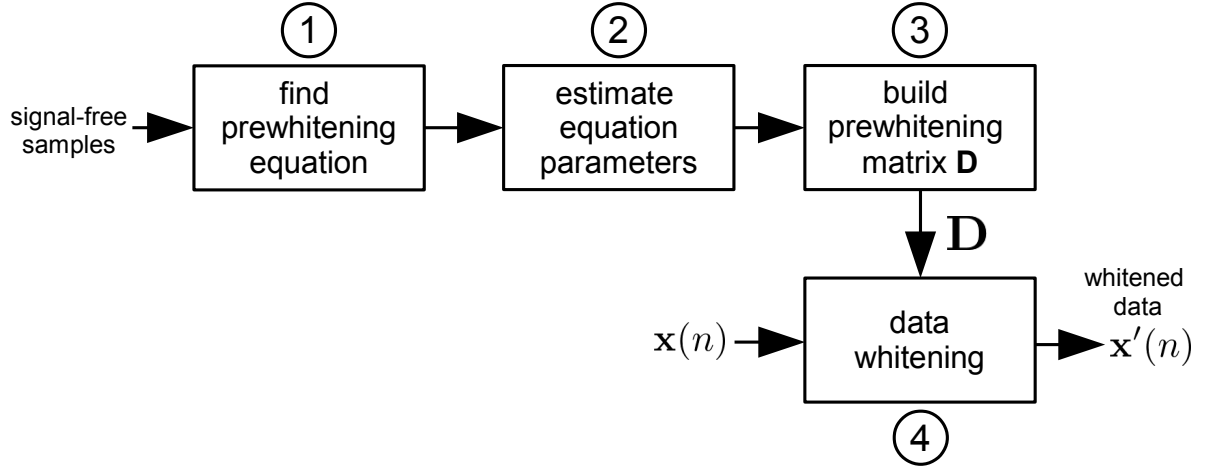


Figure 2.6: Deterministic prewhitening diagram.

structure (2.27) is known, the correlation factor \mathbf{L} is built

$$\mathbf{L}^{(k)} = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ \rho & \sqrt{1 - |\rho|^2} & 0 & & 0 \\ \rho^2 & \rho\sqrt{1 - |\rho|^2} & \sqrt{1 - |\rho|^2} & \ddots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ \rho^{M-1} & \rho^{M-2}\sqrt{1 - |\rho|^2} & \rho^{M-3}\sqrt{1 - |\rho|^2} & \dots & \sqrt{1 - |\rho|^2} \end{bmatrix}. \quad (2.28)$$

For the known correlation structure the superscript (k) is inserted. The goal is to find a matrix \mathbf{D} that will decorrelate the noise. In this work, it is assumed that the same structure of the noise as shown in [28]. Therefore, the prewhitening matrix is given by

$$\mathbf{D} = \mathbf{J}_2 \cdot -\rho \cdot \mathbf{J}_1, \quad (2.29)$$

where \mathbf{J}_2 and \mathbf{J}_1 are the selection matrices $[\mathbf{0}_{(M-1) \times 1} \quad \mathbf{I}_{M-1}]$ and $[\mathbf{I}_{M-1} \quad \mathbf{0}_{(M-1) \times 1}]$ for the last $M - 1$ sensors and the first $M - 1$ sensors, respectively. By applying the matrix \mathbf{D} on the colored noise matrix, the prewhitened noise becomes white but with a smaller power proportional to the factor $\sqrt{1 - |\rho|^2}$.

$$\mathbf{N}' = \mathbf{D} \cdot \mathbf{N}^{(c)} = \mathbf{D} \cdot \mathbf{L}^{(k)} \cdot \mathbf{N} \quad (2.30)$$

$$= \mathbf{J}_2 \cdot \mathbf{L}^{(k)} \cdot \mathbf{N} - \rho \cdot \mathbf{L}^{(k)} \cdot \mathbf{J}_1 \quad (2.31)$$

$$= \sqrt{1 - |\rho|^2} \cdot \mathbf{J}_2 \cdot \mathbf{N}. \quad (2.32)$$

Equation (2.30) shows that the noise is not only white but has its variance reduced. It

can also be verified that the Vandermonde structure of the steering vectors and the relative power of the sources are preserved [25]. The matrix \mathbf{J}_2 on the last line of (2.30) shows that one sensor is sacrificed as a cost for the dewhitening process. For applying the DP the whole data vector is multiplied by the DP matrix. This results in the prewhitened input data vector

$$\mathbf{x}'(t) = \mathbf{D} \cdot \mathbf{x}(t). \quad (2.33)$$

2.3.4 Block-wise Reduced Rank Stochastic Gradient GSC (BW-RR-SG-GSC)

In this section, an alternative way to compute the reduced rank GSC beamformer is shown by using the whole information contained in one block to estimate the expected values and correlation matrices. The objective of doing block-wisely computations is to reduce computational complexity in the prewhitening steps. The prewhitening requires the computation of the SVD, as seen in Section 2.3.1, and computing the GSC in blocks reduces the amount of SVDs to one per block.

First the CM input signal block is defined as

$$\tilde{\mathbf{X}}_m = \mathbf{X}_m \cdot \text{diag}(\mathbf{e}_m), \quad (2.34)$$

where $\mathbf{X}_m = \mathbf{X}(mN)$, $\mathbf{e}_m = [e_{\text{rr}}(N(m-1)+1), \dots, e_{\text{rr}}(mN)]^H$ and m denotes the block index. The previous definitions are used to write the estimates of the covariance matrix and expected values $\tilde{\mathbf{x}}(t)$

$$\hat{\mathbf{R}}_{\tilde{\mathbf{x}}\tilde{\mathbf{x}}} = \frac{1}{L} \tilde{\mathbf{X}}_m \tilde{\mathbf{X}}_m^H, \quad (2.35)$$

$$\hat{\mathbf{E}}\{\tilde{\mathbf{x}}\} = \frac{1}{L} \tilde{\mathbf{X}}_m \cdot \mathbf{1}_N, \quad (2.36)$$

where $\mathbf{1}_N$ is a column vector of ones of length N . Inserting the estimates into the gradient of (2.15), one can write the stochastic gradient (SG) for both \mathbf{T} and $\bar{\mathbf{w}}$:

$$\hat{\nabla}_{\mathbf{T}} J_{\text{cm}} = -\frac{2}{L} \mathbf{B} \tilde{\mathbf{X}}_m (\mathbf{e}_m + \mathbf{1}_L) \bar{\mathbf{w}}^H, \quad (2.37)$$

$$\hat{\nabla}_{\bar{\mathbf{w}}} J_{\text{cm}} = -\frac{2}{L} \mathbf{T}^H \mathbf{B} \tilde{\mathbf{X}}_m (\mathbf{e}_m + \mathbf{1}_N). \quad (2.38)$$

By defining an incremented error $\mathbf{e}_m = \mathbf{e}_m + \mathbf{1}_N$, (2.37) and (2.38) can be further simplified

to generate the BW-RR-SG-GSC update rules (2.39) and (2.40).

$$\mathbf{T}(t+1) = \mathbf{T}(t) + \mu_T \tilde{\mathbf{X}}_B \mathbf{\epsilon}_m \bar{\mathbf{w}}^H, \quad (2.39)$$

$$\bar{\mathbf{w}}(t+1) = \bar{\mathbf{w}}(t) + \mu_w \mathbf{T}^H \tilde{\mathbf{X}}_B \mathbf{\epsilon}_m, \quad (2.40)$$

where $\tilde{\mathbf{X}}_B = \mathbf{B}\tilde{\mathbf{X}}_m$. The full derivation can be found in Appendix 6.2. The VIT and prewhitening schemes depicted in Sections 2.3.1-2.3.3 work in the same manner for the block-wise algorithm. The only difference is that the whole data from the output matrix is used in the HASP and VIT-HASP solutions. The SG nomenclature is used in this technique for distinction with the sample-wise LMS algorithms types.

2.3.5 Computational Complexity Analysis

In this section, the computational complexities of the classical and proposed solutions are compared. Sums and multiplications are considered for the computation of the complexity cost. The sum of the total amount of sums and multiplications are considered flops, which is the used measure unity [38]. In previous works [84,85], tables with a number of additions and multiplications are given for various GSC adaptive algorithms. Table I shows the complexity per sample of the used algorithms computed using the criteria and methods found in [38].

The LMS-GSC and BW-RR-LMS-GSC algorithms in Table II.1 have no colored noise treatment and have a quadratic number of operations. By using a DP, one matrix vector operation is added resulting in $2M^2 - 2M$ flops, thus keeping the quadratic complexity as seen in lines DP-LMS-GSC and DP-LMS-GSC. However, when the HASP algorithms are used, the SVD and a few matrix multiplications are required. This means that the computational cost becomes cubic. The SVD alone costs $4M^2N + 22N^3$ [38].

The BW-RR-LMS-GSC alleviates the SVD cost effect. Since the filter is updated block-wisely, only one SVD per block is needed. If i_b iterations per block are set, then one should process in average i_b/N SVDs per sample. In schemes HASP-BW-RR-LMS-GSC and VIT-HASP-BW-RR-LMS-GSC in Table II.1, if $i_b = N$, the cubic factors vanish reducing drastically the computational cost.

2.4 Simulations and Results

For the simulations, a ULA with 32 elements was considered and random uniformly distributed QAM signals with unitary l^2 -norm for seven sources were generated, one relative to desired signal positioned and six relative to the interferes. The source and interferes are

Table II.1: Computational costs in flops

Algorithm	Cost
LMS-GSC Section 2.2.2	$4M^2 + 4M - 4$
RR-LMS-GSC Section 2.2.3	$8M^2 + 4M + 14Mr - 10r + 2$
DP-LMS-GSC Sections 2.2.2 and 2.3.3	$6M^2 + 2M - 4$
HASP-LMS-GSC Sections 2.2.2 and 2.3.1	$8M^2N + 2dMN + 22N^3 + 2dM + 4M^2 + 4M - 4$
VIT-HASP-LMS-GSC Sections 2.3.2 and 2.2.2	$8M^2N + 2dMN + 22N^3 + 2dM + 4M^2 + 4M - 4$
DP-RR-LMS-GSC Sections 2.2.3 and 2.3.3	$10M^2 + 2M + 14Mr - 10r + 2$
HASP-RR-LMS-GSC Section 2.2.2 and 2.3.1	$8M^2N + 22N^3 + 2dMN + 2dM + 8M^2 + 4M + 14Mr - 10r + 2$
BW-RR-LMS-GSC Section 2.3.4	$8M^2 + 8Mr + 4MN + 2M - 6r - 2N$
VIT-HASP-RR-LMS-GSC Sections 2.2.3 and 2.3.2	$8M^2N + 22N^3 + 2dMN + 2dM + 8M^2 + 4M + 14Mr - 10r + 2$
HASP-BW-RR-LMS-GSC Sections 2.3.1 and 2.3.4 i_b iterations per block	$\frac{1}{i_b} \cdot (8M^2N + 22N^3 + 2dMN + 2dM) + \frac{i_b}{N} \cdot (8M^2 + 8Mr + 4MN + 2M - 6r - 2N)$
VIT-HASP-BW-RR-LMS-GSC Sections 2.3.2 and 2.3.4 i_b iterations per block	$\frac{1}{i_b} \cdot (8M^2N + 22N^3 + 2dMN + 2dM) + \frac{i_b}{N} \cdot (8M^2 + 8Mr + 4MN + 2M - 6r - 2N)$
HASP-BW-RR-SG-GSC Sections 2.3.1 and 2.3.4 N iterations per block	$16M^2 + 22N^2 + 2dM + 2dM/N + 8Mr + 4MN + 2M - 6r - 2N$
VIT-HASP-BW-RR-LMS-GSC Sections 2.3.2 and 2.3.4 N iterations per block	$16M^2 + 22N^2 + 2dM + 2dM/N + 8Mr + 4MN + 2M - 6r - 2N$

positioned at 10° , -63° , -43° , -21° , 28° , 39° and 61° , respectively. For all simulations, 600 samples were generated. Also, white Gaussian noise is added according to (1) and is latter correlated by using the structure seen in (2.28). The correlation is set to $\rho = 0.2$ and $\rho = 0.9$ and the signal to noise ratio (SNR) is fixed at 10 dB. The reduced rank is set to $r = 10$ and the results are evaluated according to the root mean squared error (RMSE):

$$\text{RMSE}(t) = \sqrt{\text{E}\{|s(t) - \hat{s}(t)|^2\}}. \quad (2.41)$$

Table II.2 summarizes the notation used in the legends of figures of this section. It is worth noting that by default colored noise is used. When white Gaussian noise is used the term “white noise” is written after the abbreviation.

The results are shown in a Monte Carlo fashion after 1000 trials. For comparison draw the RMSE for the LMS-GSC in both white and colored noise is drawn. The Standard LMS-GSC is used since to the best of our knowledge there is no state-of-the-art beamformer

for colored noise environments. Moreover, the noise structure given by (2.28) is assumed. Previous works consider noise as directional interference [13, 68] or beamformers designed with known interference direction [61]. The step size is set to $\mu_{\text{lms}} = 0.0006$ in order to give a similar curves as in [84, 85] and yet not to diverge in colored noise scenarios. The same μ_{lms} is adopted for the other simulations in this section. This same approach is used for step size choice in the other simulations seen in this section. The optimization of μ_{lms} for colored noise scenarios is a topic for a future work. Results for the LMS-GSC algorithms are seen in Figure 2.7.

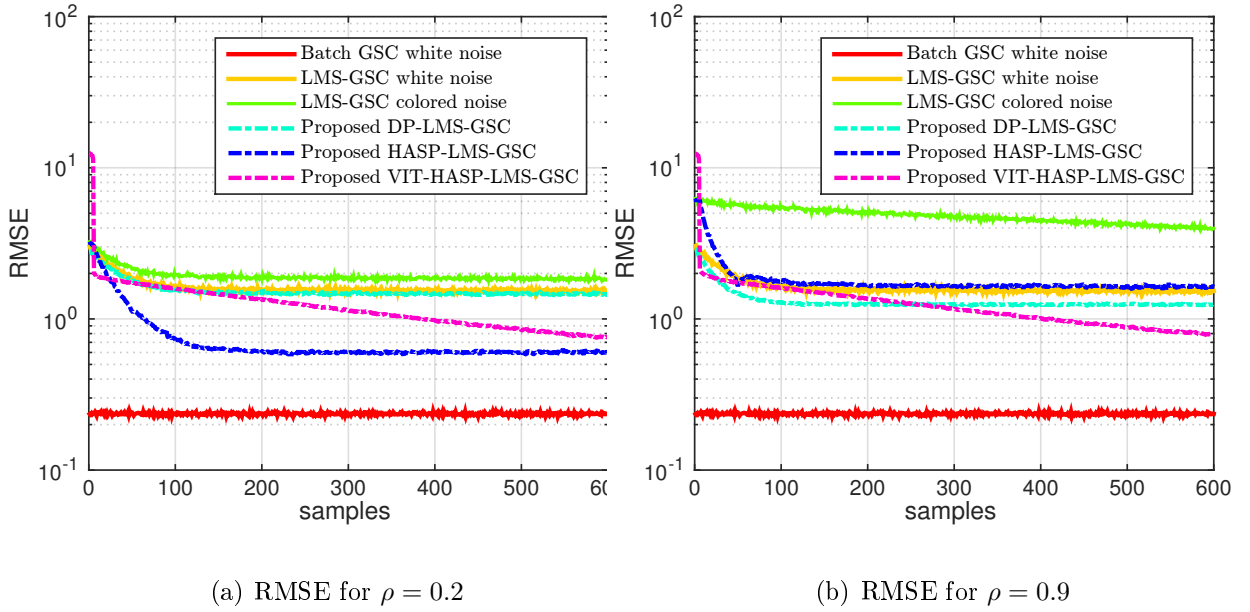


Figure 2.7: RMSE of the LMS-GSC type algorithms versus samples for $N = 50$ and an SNR of 10 dB

Note that in Figure 2.7b, the degradation of performance of the LMS-GSC white noise to the LMS-GSC. The only difference between both curves is that in LMS-GSC white noise, the noise correlation is 0 and, in LMS-GSC, the noise correlation is 0.9. If noise-only samples are available, the proposed HASP solution represented by the dark blue curve can be used. The RMSE of the HASP-LMS-GSC scheme is drastically reduced in comparison with the standard LMS-GSC. For a VIT-HASP solution an even further reduction of RMSE can be observed. For comparison, the batch GSC solution is plotted, i.e. the non-adaptive solution using all the samples to estimate beamforming filter [47]. As the system evolves in time, the proposed VIT-HASP-LMS-GSC, which run in colored noise scenario, almost reaches the batch GSC white noise curve. Applying the proposed DP-LMS-GSC for a known correlation

Table II.2: Notation of the legends used in the figures

Abbreviation	Description
Batch GSC white noise	State-of-the-art GSC using all samples to estimate \mathbf{w} in a white noise scenario
LMS-GSC white noise	State-of-the-art adaptive GSC in a white noise scenario
LMS-GSC	State-of-the-art adaptive GSC in a colored noise scenario
Proposed DP-LMS-GSC	Adaptive GSC with deterministic prewhitening in a colored noise scenario
Proposed HASP-LMS-GSC	Adaptive GSC with high accuracy stochastic prewhitening in a colored noise scenario
Proposed VIT-HASP-LMS-GSC	Adaptive GSC with Vandermonde invariance transformation based high accuracy stochastic prewhitening in a colored noise scenario
RR-LMS-GSC white noise	State-of-the-art reduced rank adaptive GSC in a white noise scenario
RR-LMS-GSC	State-of-the-art reduced rank adaptive GSC in a colored noise scenario
HASP-RR-LMS-GSC	Adaptive reduced rank GSC with high accuracy stochastic prewhitening in a colored noise scenario
VIT-HASP-RR-LMS-GSC	Adaptive reduced rank GSC with Vandermonde invariance transformation based high accuracy stochastic prewhitening in a colored noise scenario
Proposed BW-RR-LMS-GSC white noise	Block-wise reduced rank adaptive GSC in a white noise scenario
Proposed HASP-BW-RR-SG-GSC	Block-wise reduced rank adaptive GSC with high accuracy stochastic prewhitening in a colored noise scenario
Proposed VIT-HASP-BW-RR-LMS-GSC	Block-wise reduced rank adaptive GSC with Vandermonde invariance transformation based high accuracy stochastic prewhitening in a colored noise scenario

structure makes the algorithm performs better than in a white noise environment, since the noise is reduced.

In Figure 2.8, the simulations for the RR algorithms are rerun. The step sizes are set to $\mu_{\bar{w}} = 0.0001$ and $\mu_T = 0.00001$. Also the standard RR-LMS-GSC is used for comparison. For the block-wise algorithm $\mu_{\bar{w}} = 0.0001/N$ and $\mu_T = 0.00001/N$. For the VIT-HASP schemes the step sizes were slightly decreased to $\mu_{\bar{w}} = 0.00005$ and $\mu_T = 0.000008$ for sake of implementation stability. In Figure 2.8, it is also seen that even though the RR schemes have a superior overall performance, the colored noise provokes a large reduction of performance and shows a clear necessity of whitening the noise. The deterministic prewhitening has a positive, but not large, effect by decreasing RMSE. The proposed HASP-RR-LMS-GSC was shown to outperform the other methods. The combined VIT-HASP-RR-LMS-GSC has an even smaller RMSE outperforming the HASP-RR-LMS-GSC.

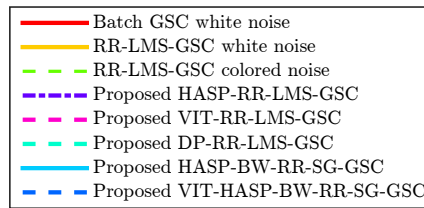
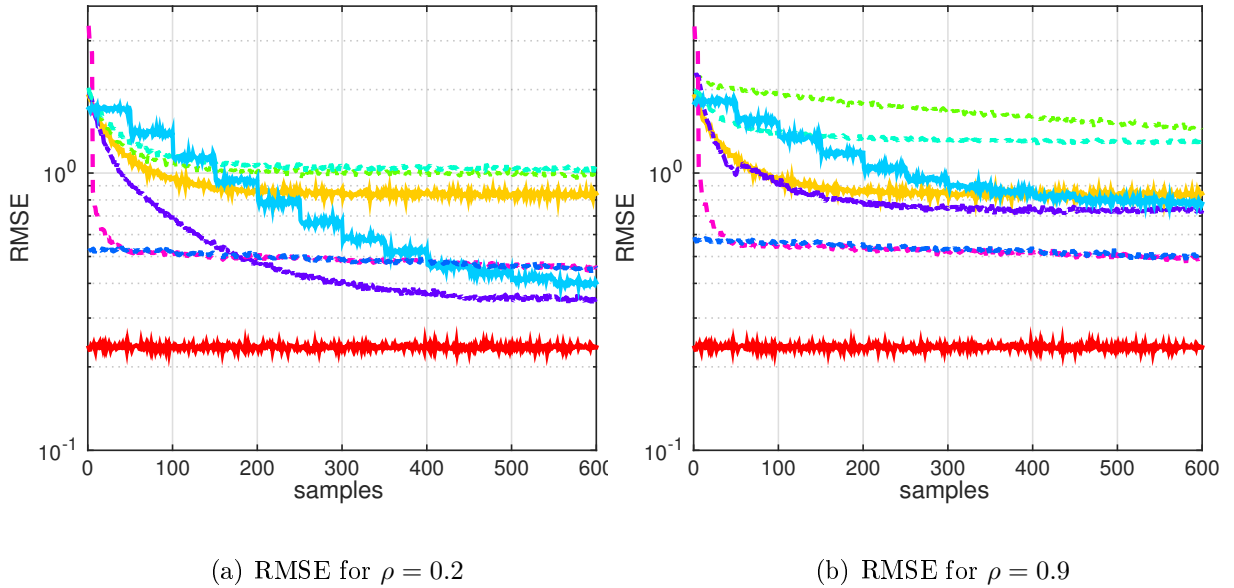


Figure 2.8: RMSE of the RR-LMS-GSC type algorithms versus samples for $N = 50$ and an SNR of 10 dB

Since the SVD increases significantly the computational complexity of the algorithms using a HASP solution, the BW-RR-SG-GSC is also shown as an alternative when computational power is not sufficiently available. The computational savings come at the cost of response time. The block size is chosen to $N = 50$ and the number of iterations within each

block is $2N$. Even though a slower convergence is noticed, the HASP-BW-RR-SG-GSC gets close to the HASP-RR-LMS-GSC on the final RMSE. Also, the VIT plays an important role in reducing the RMSE. As shown in Fig. 2.8, the VIT does most of the filtering task and the gain of the adaptive GSC becomes smaller. The steep descent seen on the first 50 samples of the VIT-HASP-RR-LMS-GSC curve is basically due the data length when performing the low rank approximation. Once the data reaches its limit, here set to 50, the RMSE remains almost constant.

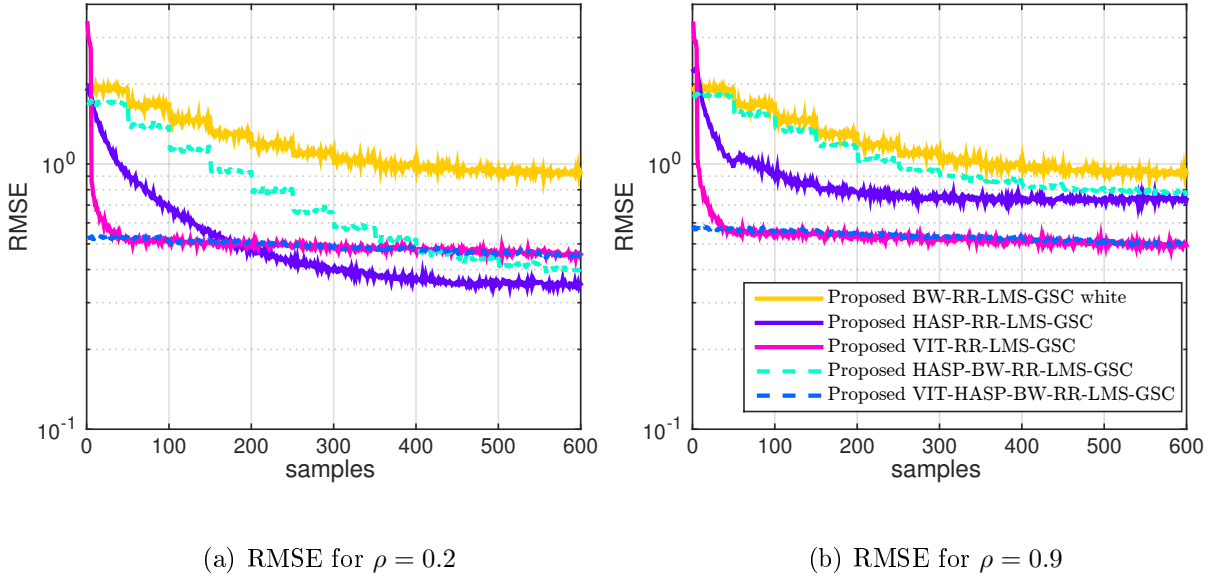
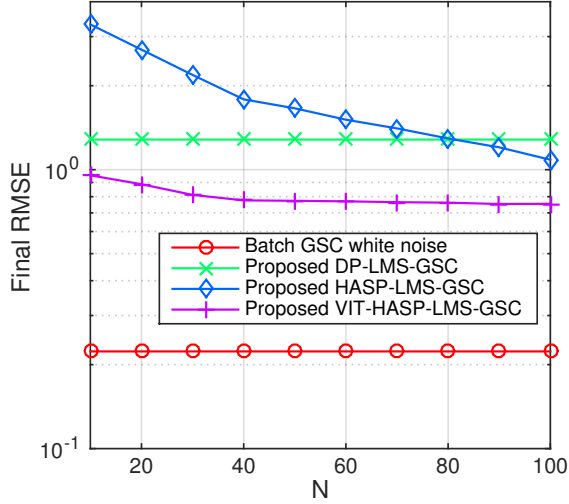


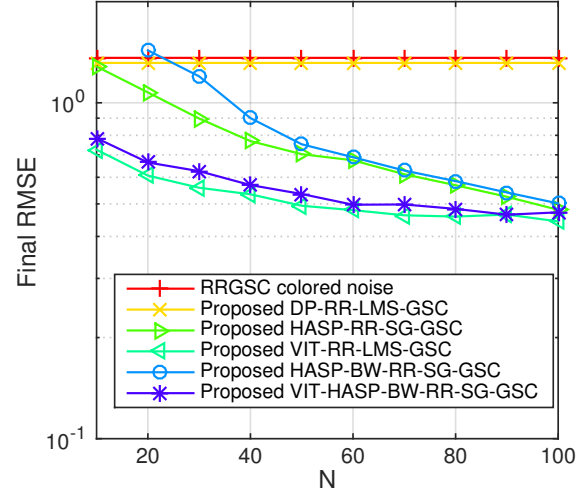
Figure 2.9: RMSE for LMS-GSC type algorithms with a varying N and an SNR of 10 dB

In Figure 2.9, the BW-RR-SG-GSC in white noise scenarios is compared with the HASP-BW-RR-LMS-GSC and the VIT-HASP-BW-RR-LMS-GSC. The HASP-RR-LMS-GSC and the VIT-HASP-RR-LMS-GSC curves are left for reference. The BW-RR-SG-GSC does not improve RMSE in a colored noise scenario with $\rho = 0.9$. In Figure 2.9, the HASP-BW-RR-SG-GSC has a similar RMSE when compared to the BW-RR-SG-GSC in a white noise environment and the VIT-HASP-BW-RR-LMS-GSC has a similar RMSE to the VIT-HASP-RR-LMS-GSC.

In Figure 2.10a the evolution of the RMSE in terms of block size is shown for the full rank algorithms. The HASP-LMS-GSC has a better RMSE than the DP-LMS-GSC when $N > 80$ and the VIT-HASP-LMS-GSC has its RMSE almost unchanged for $N > 50$. In Figure 2.10b, the evolution of the RMSE in terms of block size is shown for the reduced rank algorithms. As shown in Figure 2.10b, the block size plays an important role in the reduced rank algorithms, specially for the HASP-only prewhitening schemes. The VIT based HASP algorithms have the best performance in comparison to the other schemes. The final plots shows the final RMSE after 600 snapshots for a varying correlation ρ in Figure 2.11. Figure



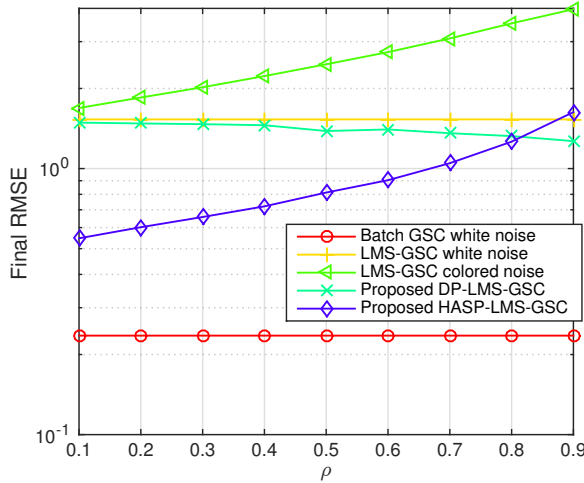
(a) RMSE for LMS type algorithms



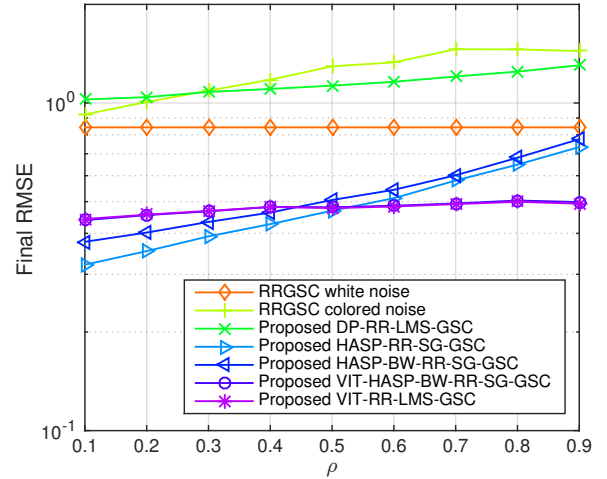
(b) RMSE for RR type algorithms

Figure 2.10: Final RMSE for algorithms with a varying N , an SNR of 10 dB and $\rho = 0.9$.

2.11a shows that the RMSE for the proposed DP-LMS-GSC remains basically constant with the increase in ρ . The proposed HASP-LMS-GSC has very low RMSE and increases with the increase of ρ , but still ending with an RMSE comparable to that of the white noise case. Figure 2.11b shows the RMSE for the reduced rank algorithms. In Figure 2.11b, it is seen



(a) RMSE for LMS type algorithms



(b) RMSE for RR-LMS type algorithms

Figure 2.11: Final RMSE after $N = 600$ snapshots with a varying ρ and an SNR of 10 dB

that besides the DP case, all the proposed schemes operate with RMSE levels below of that in the white noise case.

2.5 Summary

In this chapter, the adaptive GSC and the reduced rank adaptive GSC are extended for colored scenarios. As shown in this chapter, the colored noise degrades significantly the performance of the GSC and RR-GSC algorithms. To reduced RMSE in these scenarios, the DP-LMS-GSC, the HASP-LMS-GSC and the VIT-HASP-LMS-GSC algorithms for the full rank adaptive GSC and the DP-RR-LMS-GSC, HASP-RR-LMS-GSC and VIT-RR-LMS-GSC for the reduced rank adaptive GSC are proposed. These algorithms are based on the DP, on the stochastic prewhitening and on the VIT. The DP is used when the correlation structure of the noise is known and the stochastic prewhitening when it is unknown. The VIT acts as a pre-beamformer reducing the power that is not in direction of the desired signal.

The stochastic prewhitening requires one SVD to be computed at each sample. To reduce the number of SVD computations, a block-wise SG-RR-GSC was proposed. The algorithm adapts its filter block by block making only one SVD per block necessary reducing computational cost. The block-wise SG-RR-GSC in colored noise scenarios led to the development of the HASP-BW-RR-SG-GSC and VIT-BW-RR-SG-GSC algorithms.

The incorporation of prewhitening schemes transforms back the colored noise into white noise allowing a significant improvement of the GSC and RR-GSC. Based on that, several algorithms were proposed for colored noise scenarios. A block-wise algorithm was proposed to reduced the number of SVD computations to one per block. By means of simulation, a lower RMSE was achieved in the final result when compared to the sample-wise methods.

III

MULTIDIMENSIONAL BEAMFORMER FOR SEPARABLE ARRAYS

The tendency for a high number of antennas in 5G standards may create many practical cases in which antenna arrays have structures that can be exploited via multidimensional techniques. If the array manifold is said to be separable, i.e. it can be built from outer products of one dimensional arrays, it can be written in a natural and convenient form [73]. A practical example with this structure is the uniform rectangular array (URA). The classical approach to beamforming ignores the multidimensional nature of the data by vectorizing the array prior to applying the algorithm [4, 82].

Even though the exploitation of the tensor structure of the data reduces parameter estimation error [25, 43], there has only been very little prior work on the exploitation of multidimensionality for beamforming. In [39], a translation invariant array is considered for creating a tensor. Then, a parallel factor (PARAFAC) decomposition [54] is applied to find the filter weights in a constrained filtering problem. In this work, the generalized sidelobe canceller (GSC) [40] is used. The GSC solves a constrained problem by decoupling the constraints so that the adaptive part of the filtering is unconstrained. Moreover, the several tensor data rearrangements, unfoldings, are jointly used, thereby, exploiting the multidimensional separable structure. This breaks the solution into R smaller dimension GSCs, reducing the overall system complexity. Finally, tensor unfoldings are used to virtually increase the number of samples, thus enhancing the quality of the beamformer and creating a novel approach to beamforming, the proposed R -Dimensional GSC (R -D GSC).

This chapter is divided into six sections including this introduction. Section 3.1 shows the data model. Section 3.2 describes the classical GSC algorithm. Then, Section 3.3 presents the proposed R -D GSC for multidimensional array processing. Simulations and results are

shown in Section V. Finally, a summary is made in Section VI.

3.1 Data Model

Consider a vector $\mathbf{s}_i = [s_i(1), \dots, s_i(N)]^T$ containing the transmitted symbols $s_i(t)$ associated with the i -th source. These symbols reach the antennas of an R -dimensional array. For clarity of illustration, a uniform rectangular array (URA) is used, as depicted in Figure 3.1. Later in this section, the extension for higher dimension arrays will be made.

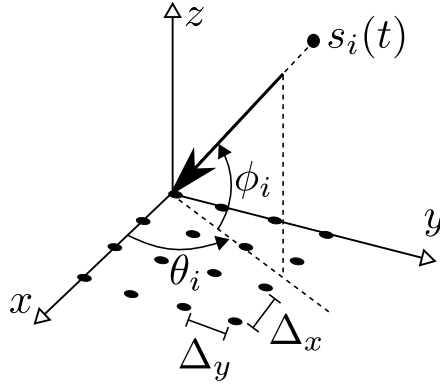


Figure 3.1: Elevation ϕ_i and azimuth θ_i of the direction of arrival of the i -th source

An impinging planar wavefront from the i -th source reaches the array with azimuth θ_i and elevation ϕ_i causing them to be received with phase-delays

$$a_i^{(m_1, m_2)} = e^{-j(m_1 \varphi_i^{(1)} + m_2 \varphi_i^{(2)})}, \quad (3.1)$$

$$\varphi_i^{(1)} = \Delta_x \cos(\theta_i) \cos(\phi_i), \quad (3.2)$$

$$\varphi_i^{(2)} = \Delta_y \sin(\theta_i) \cos(\phi_i), \quad (3.3)$$

where m_1 and m_2 are the corresponding antenna positions in the x and y directions respectively, and Δ_x and Δ_y are the inter-element spacings. The phases for a URA of size $M_1 \times M_2$ are grouped in the matrix $\mathbf{A}_i \in \mathbb{C}^{M_1 \times M_2}$ of the same size. In the presence of d sources, the received signal can be represented by

$$\mathbf{X}(t) = \sum_{i=0}^{d-1} \mathbf{A}_i s_i(t) + \mathbf{V}(t) \in \mathbb{C}^{M_1 \times M_2}, \quad (3.4)$$

where $\mathbf{V}(t)$ contains zero mean complex white Gaussian (ZMCWG) noise. As shown in Section 3.2, classical systems frequently use the vectorized form of $\mathbf{X}(t)$,

$$\mathbf{x}(t) = \text{vec}(\mathbf{X}(t)), \quad (3.5)$$

as input of the system. Therefore, the 2-Dimensional system is transformed into a unidimensional one.

To avoid vectorization and obtain a more compact notation, d steering matrices \mathbf{A}_i can be stacked along the 3rd dimension to form a steering tensor $\mathcal{A} \in \mathbb{C}^{M_1 \times M_2 \times d}$. Also, consecutive signal symbols are further grouped in the matrix $\mathbf{S} = [\mathbf{s}_0, \dots, \mathbf{s}_{d-1}]^T \in \mathbb{C}^{d \times N}$. The received signal model (3.4) can be then extended to a tensor model

$$\mathcal{X} = \mathcal{A} \times_3 \mathbf{S}^T + \mathcal{V} \in \mathbb{C}^{M_1 \times M_2 \times N}, \quad (3.6)$$

where \mathcal{V} is the $M_1 \times M_2 \times N$ noise tensor containing ZMCWG noise. Figure 3.2 illustrates the process of formation of the received tensor.

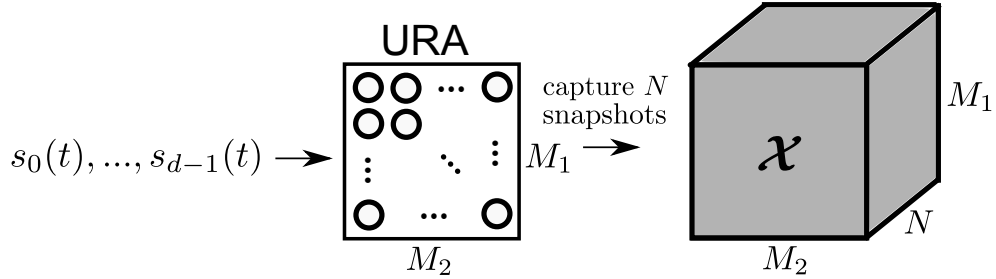


Figure 3.2: Using a URA to form a 3-D data tensor

For (3.6) to be valid the antenna array must be separable [73]. This is equivalent of saying that the steering matrix \mathbf{A}_i can be formed by the outer product between two vectors. From Equation (3.1) it is also clear that URAs have a separable structure since $\mathbf{A}_i = \mathbf{a}_i(\varphi^{(1)}) \circ \mathbf{a}_i(\varphi^{(2)})$, where $\mathbf{a}_i(\varphi^{(1)}) = [a_i^{(0,0)}, \dots, a_i^{(M_1-1,0)}]$ and $\mathbf{a}_i(\varphi^{(2)}) = [a_i^{(0,0)}, \dots, a_i^{(0,M_2-1)}]$ are the vectors formed by varying one dimension and zeroing the others. To generalize the following derivations, the data model is extended to arrays with R dimensions, i.e. $\mathcal{A}_i = \mathbf{a}_i(\varphi^{(1)}) \circ \mathbf{a}_i(\varphi^{(2)}) \circ \dots \circ \mathbf{a}_i(\varphi^{(R)})$, the received tensor model in Equation (3.6) can be further extended to

$$\mathcal{X} = \mathcal{A} \times_{R+1} \mathbf{S}^T + \mathcal{V} \in \mathbb{C}^{M_1 \times M_2 \times \dots \times M_R \times N}, \quad (3.7)$$

where the steering tensor \mathcal{A} has size $M_1 \times M_2 \times \dots \times M_R \times d$. Please note that in the case of antenna arrays, $R \leq 3$.

3.2 Batch GSC for multiple dimensions

In this section, the application of the classical batch GSC to a multiple dimension antenna array is shown. Thus, the same steps shown in Section 2.2.1 are repeated with a slight

notation change to support multidimensional data. The classical GSC [40] block diagram is depicted in Figure 3.3.

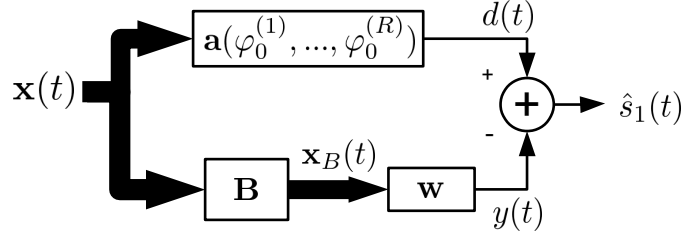


Figure 3.3: Classical GSC for R dimensions.

First, in the upper part of Figure 3.3, the signal $d(t) = \mathbf{a}^H(\varphi_0^{(1)}, \dots, \varphi_0^{(R)})\mathbf{x}(t)$ is extracted by a delay and sum filter $\mathbf{a}(\varphi_0^{(1)}, \dots, \varphi_0^{(R)})$ with taps computed in the same manner as (3.1) and steered in the direction of the signal $s_0(t)$. This signal is considered the desired signal, thus the notation $d(t)$, and the remaining $d - 1$ signals are considered interference. The phase-delays $\varphi_0^{(1)}$ are computed from the estimated direction of arrival (DOA) angles, e.g. θ_0 and ϕ_0 . For estimating the DOA the reader is referred to [43].

In the bottom part of Figure 3.3, a blocking matrix $\mathbf{B} \in \mathbb{C}^{(M_1 M_2 \dots M_{R-1}) \times M_1 M_2 \dots M_R}$, such that $\mathbf{B} \cdot \mathbf{a}(\varphi_0^{(1)}, \dots, \varphi_0^{(R)}) = \mathbf{0}$, is used to extract the interference signals $\mathbf{x}_B = \mathbf{B} \cdot \mathbf{x}(t)$. In an ideal system, the filter \mathbf{w} is computed as the Wiener solution of $|d(t) - \mathbf{w}^H \mathbf{x}_B(t)|^2$ to give an interference-removing, or sidelobe canceller, filter:

$$\mathbf{w} = \mathbf{R}_{\mathbf{x}_B}^{-1} \mathbf{r}_{\mathbf{x}_B d}, \quad (3.8)$$

where, $\mathbf{R}_{\mathbf{x}_B}$ and $\mathbf{r}_{\mathbf{x}_B d}$ are the correlation matrix of $\mathbf{x}_B(t)$ and the cross correlation vector between $\mathbf{x}_B(t)$ and $d(t)$, respectively. The total filter is

$$\mathbf{w}_{\text{gsc}} = \mathbf{a}(\varphi_0^{(1)}, \varphi_0^{(2)}, \dots, \varphi_0^{(R)}) - \mathbf{B}^H \mathbf{w}, \quad (3.9)$$

thus the upper part of Figure 3.3 corresponds to the static part of the system and the bottom part is adaptive.

For practical algorithms, the ideal Wiener solution in Equation (3.8) is not reached. Instead, batch algorithms can be used to estimate $\mathbf{R}_{\mathbf{x}_B}$ by capturing N snapshots. This is equivalent to using the $(R + 1)$ -th mode unfolding of the tensor \mathcal{X} , since

$$[\mathcal{X}]_{(R+1)} = \begin{bmatrix} \text{vec}(\mathcal{X}(:, :, \dots, 1))^T \\ \text{vec}(\mathcal{X}(:, :, \dots, 2))^T \\ \vdots \\ \text{vec}(\mathcal{X}(:, :, \dots, N))^T \end{bmatrix} \in \mathbb{C}^{N \times M_1 M_2 \dots M_R}. \quad (3.10)$$

Hence, the estimates can be computed as

$$\hat{\mathbf{R}}_{\mathbf{x}_B} = \frac{1}{N} \mathbf{B} [\boldsymbol{\mathcal{X}}]_{(R+1)}^H [\boldsymbol{\mathcal{X}}]_{(R+1)} \mathbf{B}^H, \quad (3.11)$$

$$\hat{\mathbf{r}}_{\mathbf{x}_B d} = \frac{1}{N} \mathbf{B} [\boldsymbol{\mathcal{X}}]_{(R+1)}^T \mathbf{d}^*, \quad (3.12)$$

where $\mathbf{d} = [d(1), d(2), \dots, d(N)]^T$.

3.3 Proposed R -Dimensional GSC

The classical GSC uses the vectorized form $\mathbf{x}(t)$ of $\boldsymbol{\mathcal{X}}$ and in this process it loses its multidimensional structure. To take full advantage of the tensor structure of the data, one can analyze the unfoldings of $\boldsymbol{\mathcal{X}}$. This is done to take advantage of parallel computations, smaller size matrix inversions and to virtually increase the number of samples. In the URA case, it is analyzed that the 1-st and 2-nd mode unfoldings are

$$[\boldsymbol{\mathcal{X}}]_{(1)} = [\boldsymbol{\mathcal{X}}(:, 1, 1), \dots, \boldsymbol{\mathcal{X}}(:, 1, N), \boldsymbol{\mathcal{X}}(:, 2, 1), \dots, \boldsymbol{\mathcal{X}}(:, M_2, N)], \quad (3.13)$$

$$[\boldsymbol{\mathcal{X}}]_{(2)} = [\boldsymbol{\mathcal{X}}(1, :, 1), \dots, \boldsymbol{\mathcal{X}}(M_1, :, 1), \boldsymbol{\mathcal{X}}(2, :, 2), \dots, \boldsymbol{\mathcal{X}}(M_1, :, N)]. \quad (3.14)$$

Along the columns of (3.13) and (3.14), the phase delays (3.2) and (3.3) are preserved, in that order. Therefore, the columns of (3.13) and (3.14) can be used to separately compute R components of an enhanced new filter. This enhancement is possible since the number of snapshots is virtually increased to $\sum_r NM_r$ against N snapshots in the classical GSC.

The basic concept behind the R -D GSC is to use the $R+1$ mode unfoldings of the tensor $\boldsymbol{\mathcal{X}}$ to take advantage of the tensor structure of the data. Figure 3.4 shows the R -D GSC block diagram taking advantage of this feature.

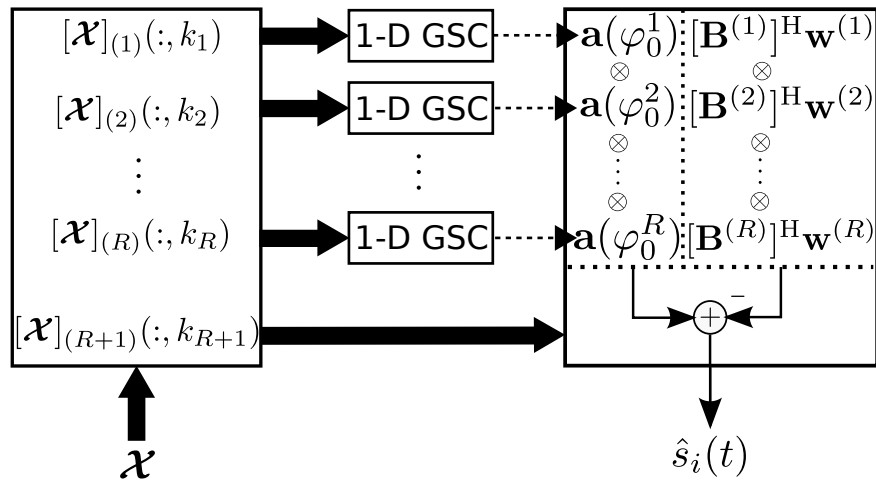


Figure 3.4: Block diagram of the R -D GSC algorithm

First, the tensor \mathcal{X} is unfolded into its $R+1$ modes. R classical GSC blocks, as explained in Section 3.2, are fed columns of the of the r -th mode unfoldings $[\mathcal{X}]_{(r)}$ indexed by $k^{(r)}$. This contrasts with the with Section 3.2, where only time samples are considered as input to the system. The 1-D GSC blocks would ideally find the Wiener solutions of $|d^{(r)}(k_r) - [\mathbf{w}^{(r)}]^H \mathbf{x}_{\mathbf{B}}^{(r)}(k_r)|^2$ for each r -th dimension:

$$\mathbf{w}^{(r)} = [\mathbf{R}^{(r)}]^{-1} \mathbf{r}^{(r)}, \quad (3.15)$$

where $\mathbf{x}_{\mathbf{B}}^{(r)}(k^{(r)}) = \mathbf{B}^{(r)}[\mathcal{X}]_{(r)}(:, k_r)$, $\mathbf{R}^{(r)}$ is the correlation matrix of $\mathbf{x}_{\mathbf{B}}^{(r)}(k^{(r)})$ and $\mathbf{r}^{(r)}$ is the crosscorrelation vector between $\mathbf{x}_{\mathbf{B}}^{(r)}(k^{(r)})$ and $d^{(r)}(k_r)$. Similarly as in Section 3.2, $d^{(r)}(k_r)$ is the output of the delay and sum filter corresponding to the desired signal extracted from the r -th mode unfolding. The total filters are computed as follows

$$\mathbf{w}_{\text{gsc}}^{(r)} = \mathbf{a}(\varphi_0^{(r)}) - [\mathbf{B}^{(r)}]^H \mathbf{w}^{(r)}. \quad (3.16)$$

To estimate the correlation matrices and the crosscorrelation vectors for the batch GSC algorithms, the unfoldings $[\mathcal{X}]_{(r)}$ are computed and then applied in the same manner as in (3.11) and (3.12) except by the Hermitian and transpose on the unfoldings:

$$\hat{\mathbf{R}}^{(r)} = \frac{1}{N} \mathbf{B}^{(r)}[\mathcal{X}]_{(r)}[\mathcal{X}]_{(r)}^H [\mathbf{B}^{(r)}]^H, \quad (3.17)$$

$$\hat{\mathbf{r}}^{(r)} = \frac{1}{N} \mathbf{B}^{(r)}[\mathcal{X}]_{(r)}[\mathbf{d}^{(r)}]^*. \quad (3.18)$$

Once the R filters are computed, they should be combined to form the final filter with length $M_1 M_2 \dots M_R$. Since a separable array is considered, the final filter computed as the Kronecker product of both static and adaptive part is proposed:

$$\mathbf{w}_{\text{rd}} = \mathbf{a}(\varphi_0^{(1)}) \otimes \mathbf{a}(\varphi_0^{(2)}) \otimes \dots \otimes \mathbf{a}(\varphi_0^{(R)}) - [\mathbf{B}^{(1)}]^H \mathbf{w}^{(1)} \otimes [\mathbf{B}^{(2)}]^H \mathbf{w}^{(2)} \otimes \dots \otimes [\mathbf{B}^{(R)}]^H \mathbf{w}^{(R)}. \quad (3.19)$$

Besides virtually increasing the number of samples, this scheme has the advantage of allowing each term $[\hat{\mathbf{R}}^{(r)}]^{-1} \hat{\mathbf{r}}^{(r)}$ to be computed separately. This gives the possibility of parallel computations in receivers with such capability, thus reducing computational time. Also, there is a computational advantage in using the multidimensional structure due the complexity of matrix inversions. The classical GSC requires the inversion of a matrix with size $M_1 M_2 \dots M_R \times M_1 M_2 \dots M_R$. By exploiting the multidimensionality of the data, the R -D GSC requires R matrix inversions of size $M_r \times M_r$.

3.4 Diagonal Loading

In order to perform beamforming and symbol estimation in antenna array systems, matrix inversion operation are typically necessary. In an antenna array system, when the number of snapshots is small in comparison to the number of antennas, the covariance matrix estimate tends to be bad conditioned. According to [60], the inversion of a bad conditioned covariance matrix causes noise (or clutter) amplification, degrading the filter quality. Diagonal loading alleviates this bad conditioning by adding a constant, or loading level, to the elements of the diagonal to the covariance matrix estimate [20]. In massive MIMO scenarios, such noise amplification becomes even stronger due to the small number of snapshots in comparison with the amount of antennas

The diagonally loaded correlation matrix is created by summing a diagonal matrix to the correlation matrix [36]

$$\mathbf{R}_L = \mathbf{R} + \mathbf{I} \cdot l, \quad (3.20)$$

where \mathbf{I} is the identity matrix, \mathbf{R} is the correlation matrix and l is the loading level. The load reduces the eigenvalue spread by acting as if power was added impinging from every direction. To analyze the diagonal loading effect on the beam shape, the spatial response of the computed filter is computed after diagonal loading is applied:

$$P(\theta, \phi) = |\mathbf{w}_{\text{gsc}}^H \mathbf{a}(\theta, \phi)|. \quad (3.21)$$

The effect of diagonal loading on the GSC's spatial power response is shown in Figure 3.5.

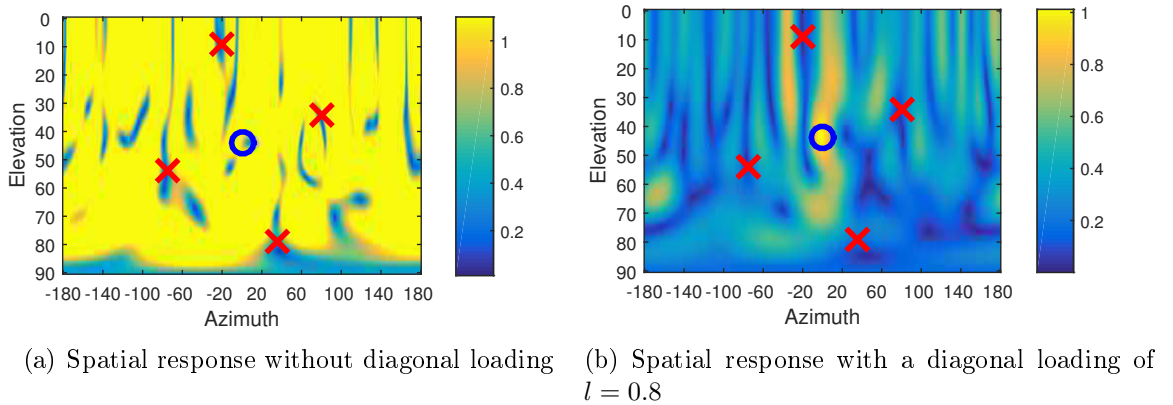


Figure 3.5: Magnitude of the spatial response of the GSC filter

In Figure 3.5a, it is seen that the filter succeeds in reducing power from the interference. The constraint also sets $P(\theta, \phi) = 1$ in the desired signal direction. However, the beamformer filter spatial power response is high in the directions where no source is present what can degrade signal quality. Moreover, the global maximum is not at the desired signal direction

what can cause thermal noise amplification. Figure 3.5b shows $P(\theta, \phi)$ after a diagonal loading is applied. In Figure 3.5b the spatial power response is as expected with a maximum at the desired signal direction, very low power in the interferences directions and low power elsewhere.

As long as the loading level is considerably lower the sources related eigenvalues in the correlation matrix, the loading should not greatly affect the systems capability of mitigating interference [36]. This concludes that diagonal loadings can be used when the desired signal quality is not as expected and the spatial power response is similar to that shown in Figure 3.5a. Therefore, this technique will be used in Section 3.5.

3.5 Simulations and Results

For the simulations, a URA with size 8×8 is used and QPSK samples are created. Then, 5 uncorrelated signals with unitary variance are inserted into the simulation scenario: one desired and 4 interferers at $(\theta_i, \phi_i) \in \{(0^\circ, 45^\circ); (-75^\circ, 55^\circ); (35^\circ, 80^\circ); (-20^\circ, 10^\circ); (80^\circ, 35^\circ)\}$. The desired signal DOA (θ_0, ϕ_0) is considered known in order find $\varphi_0^{(1)}$ and $\varphi_0^{(2)}$ for the computation of both Classic and R -D GSC approaches. Noise is added and the signal to noise ratio (SNR) is set to 0 dB. To verify the algorithms the signal to interference and noise ratio (SINR) for uncorrelated signals with unitary variance is computed as:

$$\text{SINR} = \frac{\mathbf{w}_{\text{rd}}^H [\mathcal{A}]_{(3)}(:, 1) [\mathcal{A}]_{(3)}^H(:, 1) \mathbf{w}_{\text{rd}}}{\mathbf{w}_{\text{rd}}^H ([\mathcal{A}]_{(3)}(:, 2:d) [\mathcal{A}]_{(3)}^H(:, 2:d) + \sigma_v^2 \mathbf{I}) \mathbf{w}_{\text{rd}}}, \quad (3.22)$$

where σ_v^2 is the noise variance. Diagonal loading was applied [20] to $\hat{\mathbf{R}}_{\mathbf{x}_B}$ and $\hat{\mathbf{R}}^{(r)}$ with a loading level $l = 0.8$ for the GSC and $l = \sqrt{0.8}$ for each R -D GSC correlation matrix.

The plot of the SINR for the classical GSC and R -D GSC is shown in Figure 3.6 with a varying number of samples N . The classical GSC shows a low SINR level for small N , while the R -D GSC is above 8dB at $N = 20$.

Then, the same scenario as previously is kept and the amount of samples is fixed to $N = 100$ and the SNR is varied from 0 to 60 dB. The result is shown in Figure 3.7. Figure 3.7 shows that the SINR is better for the R -D GSC when the SNR is very or high and about the same between 10 dB and 20 dB.

For the third simulation, the previous scenario is kept and the number of samples is fixed to $N = 50$ samples. Then, the size of the array is varied keeping the dimensions with same length $M_1 = M_2$. The results are shown in Figure 3.8. For this simulation, the classical GSC SINR starts at 7.6 dB and settles roughly at 12 dB as the number of antennas is increased. For the R -D GSC, since the dimensions of the final filter \mathbf{w}_{rd} are computed separately, the number of degrees of freedom is limited to the $M_r - 1$ for each dimension.

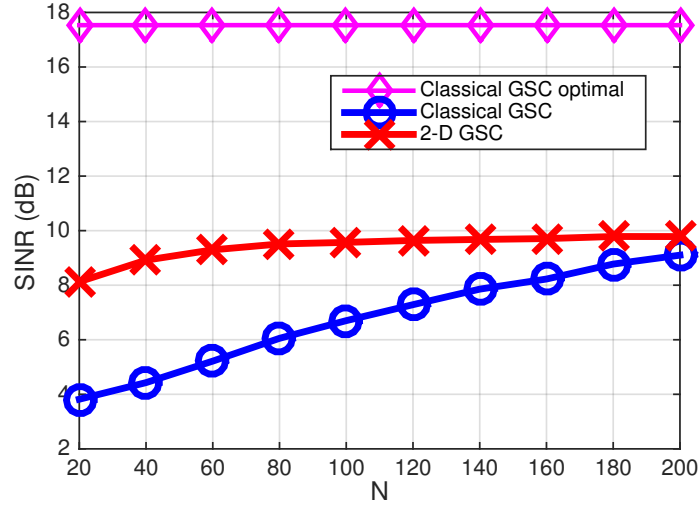


Figure 3.6: Effective SINR after applying the Classical and proposed 2-D GSC for a varying number of temporal snapshots N

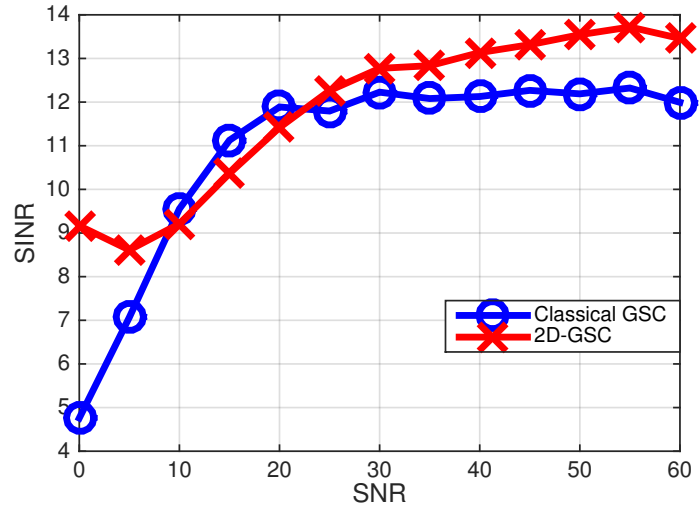


Figure 3.7: Effective SINR after applying the Classical and proposed 2-D GSC for a varying SNR

This means that the algorithm properly attenuates the interference sources when $M_r > d$, as observed in Figure 3.8. For $M_1 = M_2 \leq 5$, the R -D GSC SINR is below 8 dB. However, for $M_1 = M_2 > 5$, the R -D GSC can reach SINRs of more than 20 dB.

To verify the computational time difference between both schemes, the same scenario is kept and N is varied. Also two cases are considered. One when the unfoldings are already accomplished by hardware with such a capability and a second where the unfoldings are computed via software. The results are depicted in Figure 3.9.

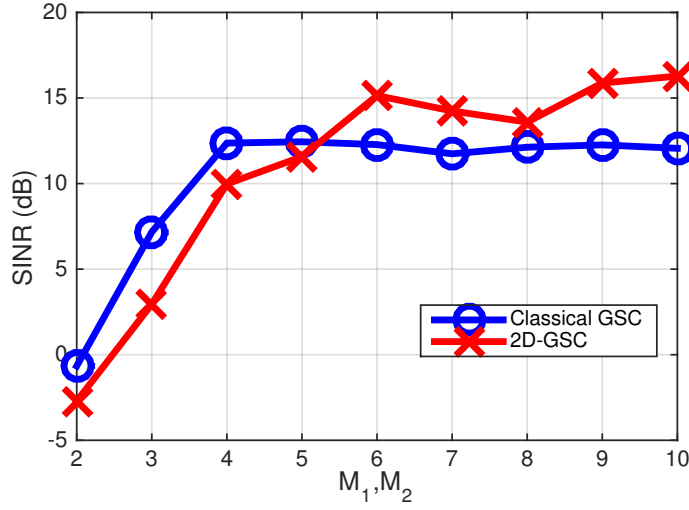


Figure 3.8: Effective SINR after applying the Classical and proposed 2-D GSC for a varying array size. Both dimensions lengths are kept equal

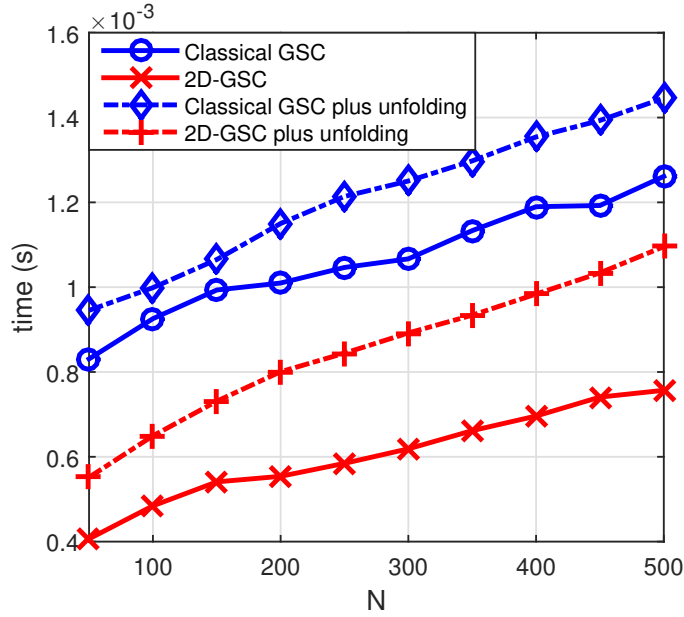


Figure 3.9: Computational time versus N

In Figure 3.9, the dashed curves show the computational time considering that the unfoldings are computed via software. The R -D GSC shows a mean computational time 30% faster than the Classical GSC. Considering that the samples are delivered already unfolded by the hardware, the R -D GSC shows a mean computational time 43% lower than the Classical GSC.

3.6 Summary

As the number of antennas and array dimensions at the receivers increase, the possibility of using multidimensional schemes arises. This chapter shows how to take advantage of this multidimensionality in order to enhance the performance of beamformers. Also, by usage of R mode unfoldings allows for parallel computations of reduced size matrix inversions, giving a lower computational complexity to the system. With this goal, the R -Dimensional GSC was presented. This scheme uses R mode unfoldings to take advantage of the multidimensionality and the simulation results show the SINR with a varying the number of samples and number of antenna elements. The results show SINR increases significantly when the number of antennas in each dimension is higher than the number of source signals.

In the future the limitation due to the reduced degrees of freedom, when the number of antennas in each dimension is lower than the number of source signals, should be addressed. Also, the R -D GSC should be extended to its adaptive form giving more flexibility and tracking capability to the beamformer.

IV

BROADBAND BEAMFORMING VIA FREQUENCY INVARIANCE TRANSFORMATION AND PARAFAC DECOMPOSITION TRACKING

Frequently, beamformers assume narrowband signals. However, the adoption of cognitive radio implies into the opportunistic exploitation of several frequency slots requiring a broadband beamforming. Moreover, millimeter wave communication systems should also deal with broadband signals. Finally, MIMO-OFDM systems assume only a small amount of OFDM channels so that the channel can be assumed as flat. Therefore, broadband beamforming can be incorporated into MIMO-OFDM systems in order to increase their throughput taking into account a huge amount of frequency slots. In the broadband scenario, the time delays cannot be considered as pure phase delays, but as time delays represented by time convolutions as it is described in the signal model in Section 4.1. Techniques designed for narrowband signals are not suitable for broadband scenarios. Therefore, bank of frequency invariant beamformers (FIBs) [76] is applied for broadband scenarios. The output signals of such bank of FIBs are represented as instantaneous mixtures, and, as a consequence, simpler separation techniques for instantaneous mixtures can be used.

This chapter proposes a framework composed of a bank of FIBs [65], combined with the PARAFAC tensor decompositions [46, 54]. Moreover, the framework exploits the adaptive PARAFAC recursive least squares tracking (RSLT) technique [70] in order to adaptively track the PARAFAC decomposition.

The original PARAFAC recursive least squares tracking (RLST) technique proposed in [70] was designed to exploit a time growing tensor, requiring the sources to be non-stationary. In this chapter, the correlation tensor is built in a novel cyclical manner, so that

the PARAFAC decomposition for non-white sources [57] can also be applied adaptively. Finally, a recursive pseudoinversion of the estimated steering matrix is proposed via the Ben-Israel and Cohen iterative pseudoinversion method [16] and it is compared with the widely used minimum mean squared error-based algorithm proposed by Frost [35].

4.1 Broadband Signal Model

Figure 4.1 illustrates a scenario containing two radio base stations equipped with antenna arrays such as those found in Chapters II and III. Each antenna receives a combination of

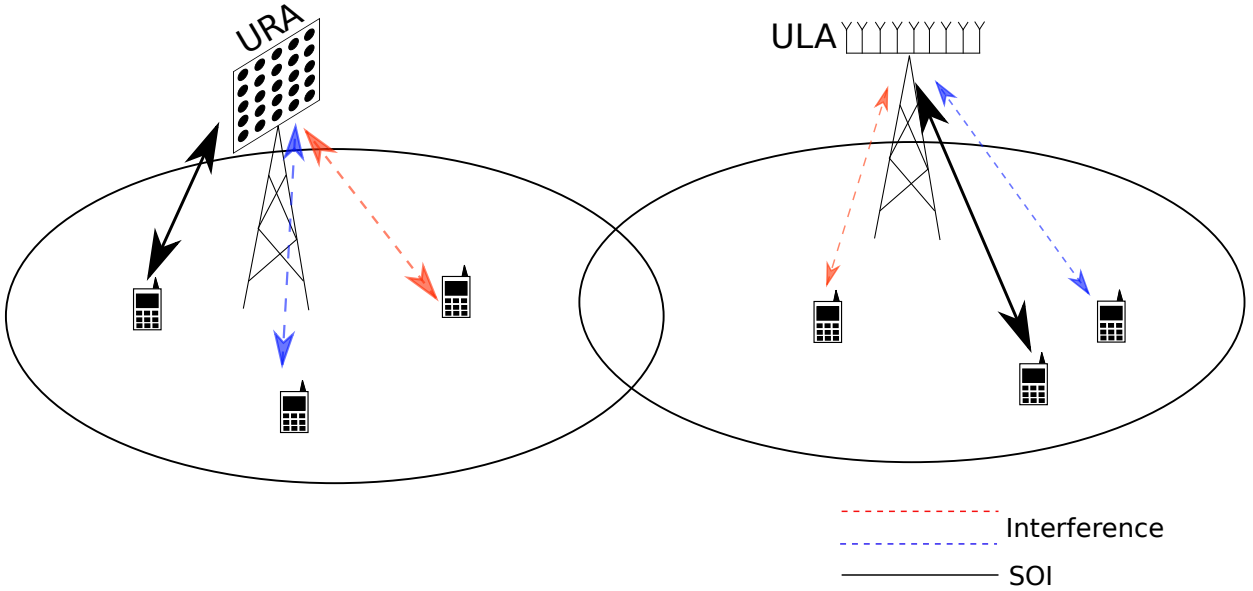


Figure 4.1: Scenario containing a URA and a ULA. Each signal has a single path to each antenna.

the signal of interest (SOI) and the interference. The signals have bandwidth B and, since their directions of arrival are different, their relative delays are different for each antenna. Therefore, the received signal at the m -th antenna can be written as a sum of time-delayed signals

$$x_m(t) = \sum_{i=0}^{d-1} s_i(t - \tau_{i,m}) + v_m(t), \quad (4.1)$$

where $\tau_{i,m}$ is the propagation delay between source i and the antenna m and $v_m(t)$ is additive Gaussian white noise. The received signal can be considered as a filtered version of the

original signals, where the filters represent pure delays

$$x_m(t) = \sum_{i=0}^{d-1} s_i(t) * \delta(t - \tau_{i,m}) + v_m(t), \quad (4.2)$$

where $*$ denotes the convolution operator and $\delta(t = 0) = 1$, while $\delta(t \neq 0) = 0$. Therefore, the difference between (4.2) and the models on Chapters II and III is that the delay is caused by the convolution with $\delta(t - \tau)$ instead of the multiplication by a phase shifting term.

When designing beamformers, the goal is to find an estimate $\hat{s}_i(t)$ for each $s_i(t)$ source at the outputs $y_i(t)$ of the beamformer. Simple narrowband models regard the propagation delay as a phase delay and, therefore, should be valid only for a specified frequency. Since in this case the source signal has a band $B = [\omega_{\min}, \omega_{\max}]$, all frequencies within B should be taken into consideration. One way to achieve this in the digital domain is to approximate the convolution by a discrete filtering process. The filtering is achieved using tap-delay lines as shown in Figure 4.2.

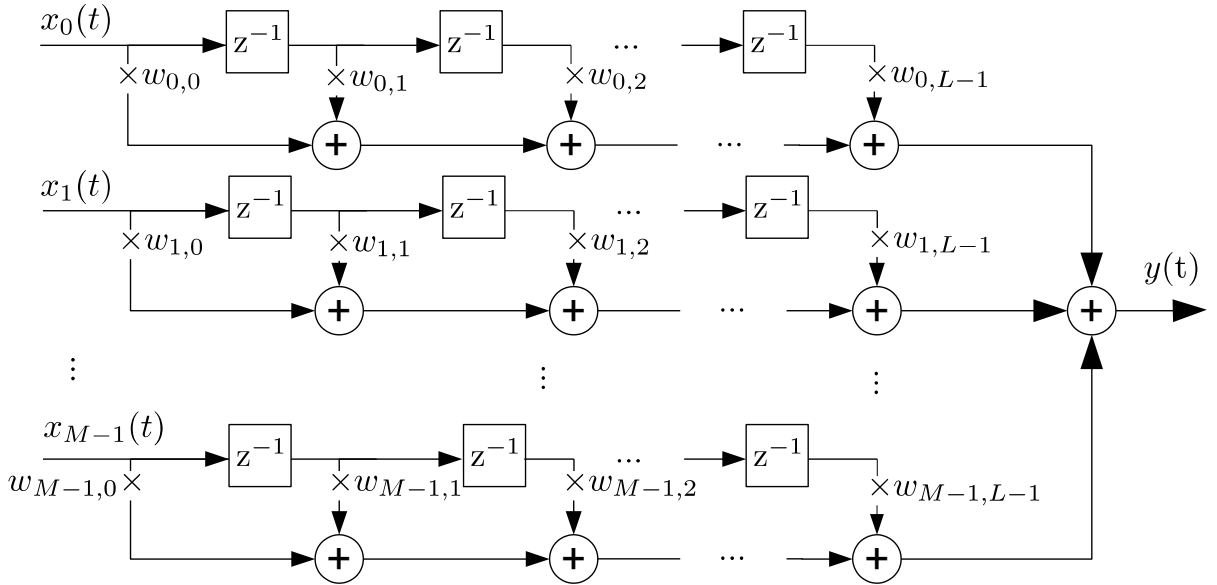


Figure 4.2: Tap-delay line. Each antenna m receives a broadband signal filtered by an FIR filter containing L taps.

For each sensor and delay, a group of weights $w_{m,l}$, for $l = 0, \dots, L - 1$ and $M = 0, \dots, M - 1$, creates M finite impulse response (FIR) filters of length L . With the weights, the reconstruction of the original signals can be attempted

$$y_i(t) = \sum_{m=0}^{M-1} \sum_{l=0}^{L-1} x_m(t - lT_s) \times w_{m,l}, \quad (4.3)$$

where T_s is the sampling rate, M is the number of antennas and L is the total length of the delay line. To allow for a compact notation, a received signal matrix is introduced as

$$\mathbf{X}(t) = \begin{bmatrix} x_0(t) & x_0(t - T_s) & \dots & x_0(t - (L - 1)T_s) \\ x_1(t) & x_1(t - T_s) & \dots & x_1(t - (L - 1)T_s) \\ x_2(t) & x_2(t - T_s) & \dots & x_2(t - (L - 1)T_s) \\ \vdots & \vdots & \ddots & \vdots \\ x_{M-1}(t) & x_{M-1}(t - T_s) & \dots & x_{M-1}(t - (L - 1)T_s) \end{bmatrix} \in \mathbb{C}^{M \times J}. \quad (4.4)$$

Even though the notation in Equation (4.4) has a more natural form since it clearly distinguishes between the space and time-delay dimensions, a vector notation is usually more practical where, instead, a received vector $\mathbf{x}(t) = \text{vec}(\mathbf{X}(t))$. The operator $\text{vec}(\cdot)$ vectorizes its argument by stacking its columns. In a similar manner, a vector \mathbf{w}_i containing all filter coefficients $w_{m,l}$ is created to write the compact form of Equation (4.3).

$$y_i(t) = \mathbf{w}_i^H \mathbf{x}(t). \quad (4.5)$$

From the frequency domain representation, Equation (4.2) can be rewritten by regarding the propagation delays as a different phase delay for each frequency bin

$$X_{m,l}(\omega) = \sum_{i=0}^{d-1} a_{m,l}(\mathbf{p}_i, \omega) S_i(\omega) + V_m(\omega), \quad (4.6)$$

where ω is the angular frequency and the array is at the origin of the coordinate system, the i -th source is at position \mathbf{p}_i and $a_{m,l}(\mathbf{p}_i, \omega)$ are elements of the steering vector defined as

$$\mathbf{a}(\mathbf{p}_i, \omega) = \text{vec}(\mathbf{A}(\mathbf{p}_i, \omega)) \in \mathbb{C}^{ML \times 1}, \quad (4.7)$$

$$\mathbf{A}(\mathbf{p}_i, \omega) = \begin{bmatrix} e^{-j\omega\tau_{i,0}} & e^{-j\omega(\tau_{i,0}+T_s)} & \dots & e^{-j\omega(\tau_{i,0}+(L-1)T_s)} \\ e^{-j\omega\tau_{i,1}} & e^{-j\omega(\tau_{i,1}+T_s)} & \dots & e^{-j\omega(\tau_{i,1}+(L-1)T_s)} \\ \vdots & \vdots & \ddots & \vdots \\ e^{-j\omega\tau_{i,M-1}} & e^{-j\omega(\tau_{i,M-1}+T_s)} & \dots & e^{-j\omega(\tau_{i,M-1}+(L-1)T_s)} \end{bmatrix} \in \mathbb{C}^{M \times L}. \quad (4.8)$$

For a ULA, $\mathbf{a}(\mathbf{p}_i, \omega)$ is dependent on the unidimensional DOA θ_i instead of the three dimensional coordinate \mathbf{p}_i in such a way that $\omega\tau_m = m\Delta_x\lambda^{-1}2\pi\sin\theta$. For generalization of the central frequency one may also use the normalized frequency $\Omega = \omega T_s$ and make the

substitution $\mu = \Delta_x / (cT_s)$ to rewrite Equation (4.8) as

$$\mathbf{A}(\theta_i, \omega) = \begin{bmatrix} 1 & e^{-j\Omega} & \dots & e^{-j\Omega(J-1)} \\ e^{-j\mu\Omega \sin \theta_i} & e^{-j\Omega(\mu \sin \theta_i + 1)} & \dots & e^{-j\Omega(\mu \sin \theta_i + (L-1))} \\ \vdots & \vdots & \ddots & \vdots \\ e^{-j\mu\Omega \sin \theta_i(M-1)} & e^{-j\mu\Omega(\sin \theta_i(M-1) + 1)} & \dots & e^{-j\mu\Omega(\sin \theta_i(M-1) + (L-1))} \end{bmatrix} \in \mathbb{C}^{M \times L}. \quad (4.9)$$

Even though Equations (4.7) and (4.8) are not practical for the creation of simulation scenarios, they are useful for the computation of the spatial and frequency response of the designed filter \mathbf{w} :

$$P(\theta, \omega) = \mathbf{w}^H \mathbf{a}(\theta, \omega). \quad (4.10)$$

As an example, let us consider the response of the filter $\mathbf{w} = \frac{1}{4}[0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0]^T$ in Figure 4.3 for $M = 4$ and $L = 5$. To obey the Nyquist theorem, $T_s = 0.5T_{\min}$ and $\Delta_x = 0.5\lambda_{\min}$, where T_{\min} is the period of the highest frequency component and λ_{\min} its wavelength, thus $\mu = 1$. Figure 4.3 clearly shows the frequency dependency

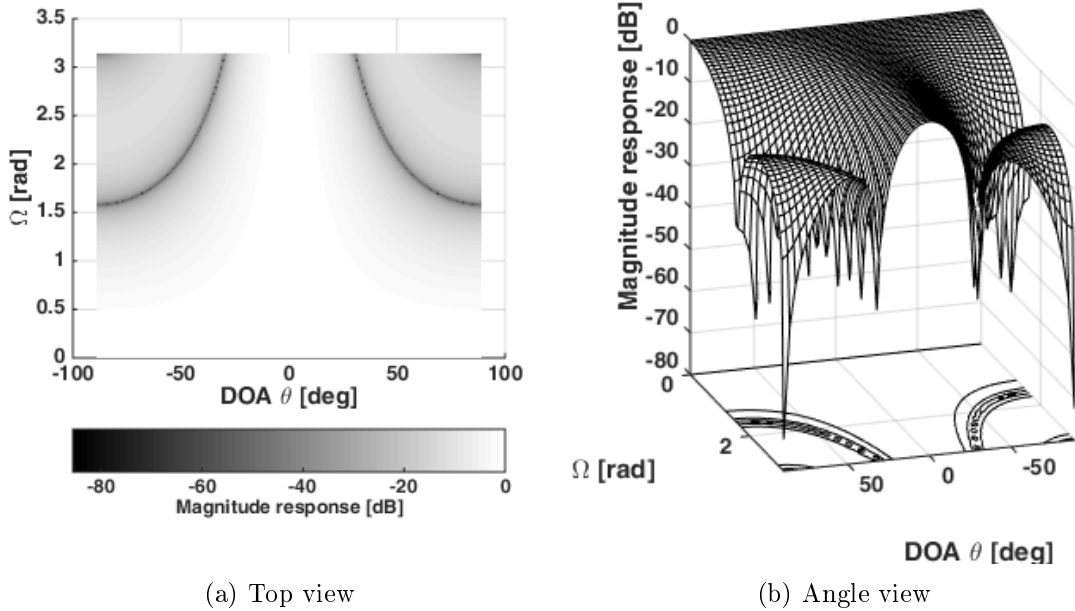


Figure 4.3: Spatial and frequency response of \mathbf{w} steered to 0°

of the broadband beamformer. Section 4.2 reviews a technique for the construction of frequency invariant beamformers.

4.2 Multi-Dimensional Inverse Fourier Transform Frequency Invariant Beamformer (FIB) Design

In this section, the work first proposed in [76] that is based on the bidimensional inverse DFT (IDFT2) is reviewed. Detailed extensions for more dimensions can be found in [64]. To review the method in [76] let us first consider the Fourier transform-based response of a continuous space and time beamformer $w(x, t)$ for and ULA

$$P(\theta, \omega) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} w(x, t) e^{-j\omega \frac{1}{c} \sin \theta x} e^{-j\omega t} dx dt, \quad (4.11)$$

where c is the wave speed and x represents sensor location along a linear axis. The basic idea is to design filter $w(x, t)$ by defining a frequency invariant response $P(\theta, \omega)$ and take its bidimensional inverse Fourier transform to find the desired tap-delay weights [76]. The process can be facilitated by the substitutions

$$\omega_1 = \omega c^{-1} \sin \theta \quad (4.12)$$

$$\omega_2 = \omega \quad (4.13)$$

in order to find the spatial-temporal spectrum that allows for the application of the inverse bidimensional Fourier transform:

$$P(\omega_1, \omega_2) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} w(x, t) e^{-j\omega_1 x} e^{-j\omega_2 t} dx dt. \quad (4.14)$$

By analyzing the equality formed from (4.12) and (4.13) $\omega_1 = \omega_2 c^{-1} \sin \theta$, it is easily verified that

$$-\frac{1}{c} \leq \frac{\omega_1}{\omega_2} \leq \frac{1}{c}. \quad (4.15)$$

Given that a signal is band limited at maximum frequency of ω_{\max} , one may further limit the spatio-temporal spectrum to the region $\omega_2 \leq \omega_{\max}$. Also, the dependency on ω has to be eliminated so that the response is frequency independent. This independence can be obtained when the spectrum obeys the relation $P(\omega_1, \omega_2) = P(c\omega_1/\omega_2)$. This results in $c\omega_1/\omega_2 = \sin \theta$, thus, asserting that the spatio-temporal spectrum is only dependent on θ .

For the discrete case, the frequency is replaced by the normalized frequency $\Omega = \omega T_s$ and the integral by discrete sums

$$P(\theta, \Omega) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} w[m, n] e^{-j\Omega m \mu \sin \theta} e^{-j\Omega n}, \quad (4.16)$$

where m is the antenna index, n is the discrete time index and the brackets $[\cdot]$ are used to point out the discrete function. In the same manner as in (4.14), substitutions are made for $\Omega_1 = \Omega\mu \sin \theta$ and $\Omega_2 = \Omega$ leading to the bidimensional DFT form of the spatial-temporal response

$$P(\Omega_1, \Omega_2) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} w[m, n] e^{-j\Omega_1 m} e^{-j\Omega_2 n}. \quad (4.17)$$

By choosing $\Delta_x = 0.5\lambda_{\min}$ and $T_s = 0.5T_{\min}$, it is found that $\mu = 1$ and $\Omega_{\max} = \pi$. Therefore, condition (4.15) reduces to $-1 \leq (\Omega_1/\Omega_2) \leq 1$. This region of existence is depicted in Figure 4.4.

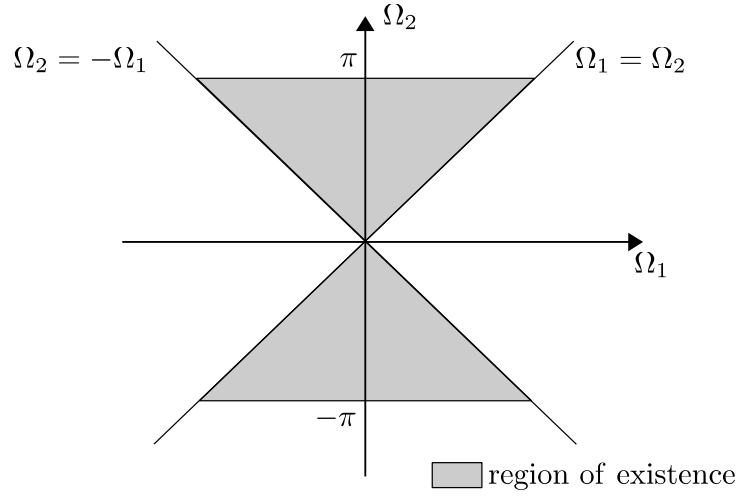


Figure 4.4: Region of existence of the spatio-temporal response

The final desired spatial-temporal response of the beamformer is

$$P(\Omega_1, \Omega_2) = \begin{cases} F(\Omega_1/\Omega_2) & \text{if } |\Omega_1/\Omega_2| \leq 1, \\ \alpha(\Omega_1, \Omega_2) & \text{if } |\Omega_1/\Omega_2| > 1, \\ \alpha(0, 0) & \text{if } \Omega_1 = \Omega_2 = 0. \end{cases} \quad (4.18)$$

A common choice for $F(\Omega_1/\Omega_2)$ is the narrowband delay and sum filter [82] as a prototype filter:

$$F(\Omega_1/\Omega_2) = \frac{1}{M_{\text{prot}}} \sum_{m=-\lceil M_{\text{prot}}/2 \rceil}^{\lfloor M_{\text{prot}}/2 \rfloor} e^{-jm\pi \sin \theta}, \quad (4.19)$$

where $\lceil \cdot \rceil$ is the rounding operator to the biggest neighbor integer and $\lfloor \cdot \rfloor$ is the rounding operator to the smallest neighbor integer. Since $\alpha(\Omega_1, \Omega_2)$ lies outside the region of existence of $P(\Omega_1, \Omega_2)$, it can be chosen arbitrarily. At the point $\Omega_1 = \Omega_2 = 0$, the DC component,

which is usually removed, is found and it carries no spatial information, therefore, it can be set to zero. The resultant beamformer is the bidimensional inverse discrete Fourier transform (IDFT2) of $P(\Omega_1, \Omega_2)$ sampled along Ω_1, Ω_2 and set to proper values of $\alpha(\Omega_1, \Omega_2)$ outside the region of existence.

As an example, the $\text{IDFT2}(P(\Omega_1, \Omega_2))$ of size 32×32 is computed for $M = 10$ and $L = 20$ and set $\alpha(\Omega_1, \Omega_2) = 0$. The beamformer coefficients $w[m, n]$ are obtained via translation and truncation of the 32×32 result of the IDFT2 to the desired $M \times L$ size. After assigning the coefficients properly inside the vector \mathbf{w} , the response is computed using Equation (4.10). The result is shown in Figure 4.5.

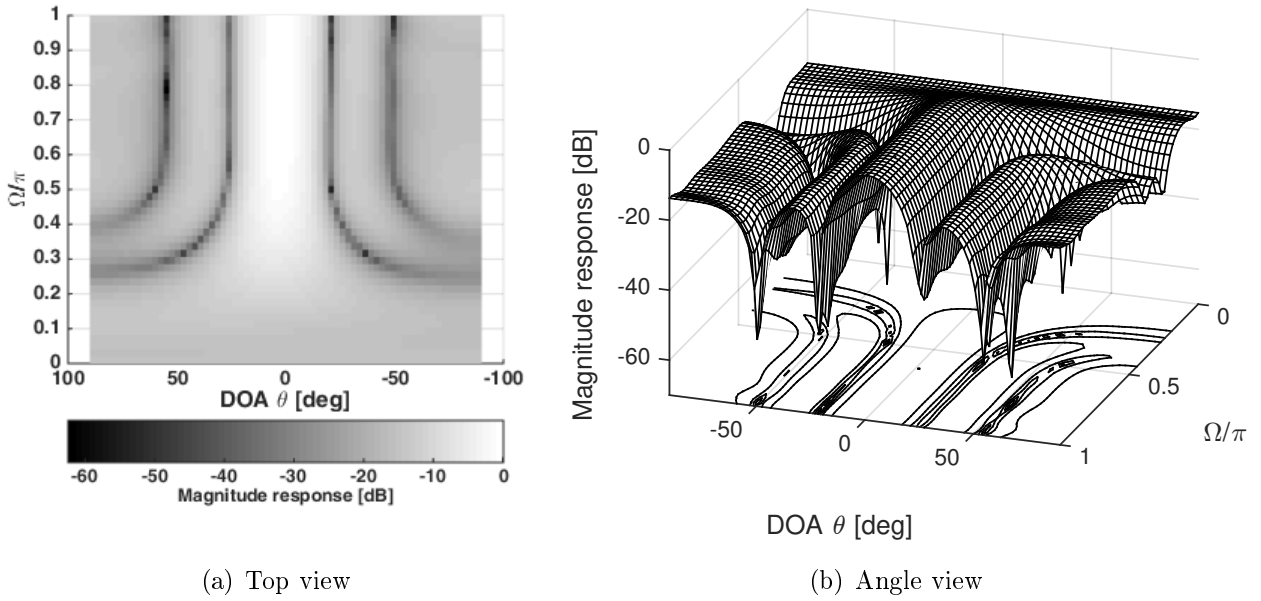


Figure 4.5: Spatial and frequency response of a FIB

In Figure 4.5, it is verified that the frequency independence of the designed filter up to $\Omega = 0.4\pi$. For reference, the coefficients are shown in Figure 4.6.

4.3 Bank of FIBs as a Virtual Array of Instantaneous Mixtures

By using the concepts in Section 4.2, it is shown how a bank of FIBs can be created for spatial filtering of broadband signals. By creating a bank of FIBs that has independent FIB responses, one can use the output from each FIB as virtual array element, where the frequency dependency, i.e. broadband characteristics, is removed. Hence, the output from the bank of FIBs can be considered as a virtual array containing instantaneous mixtures instead of convolutive ones [63–65]. To elaborate this model, the two responses shown in

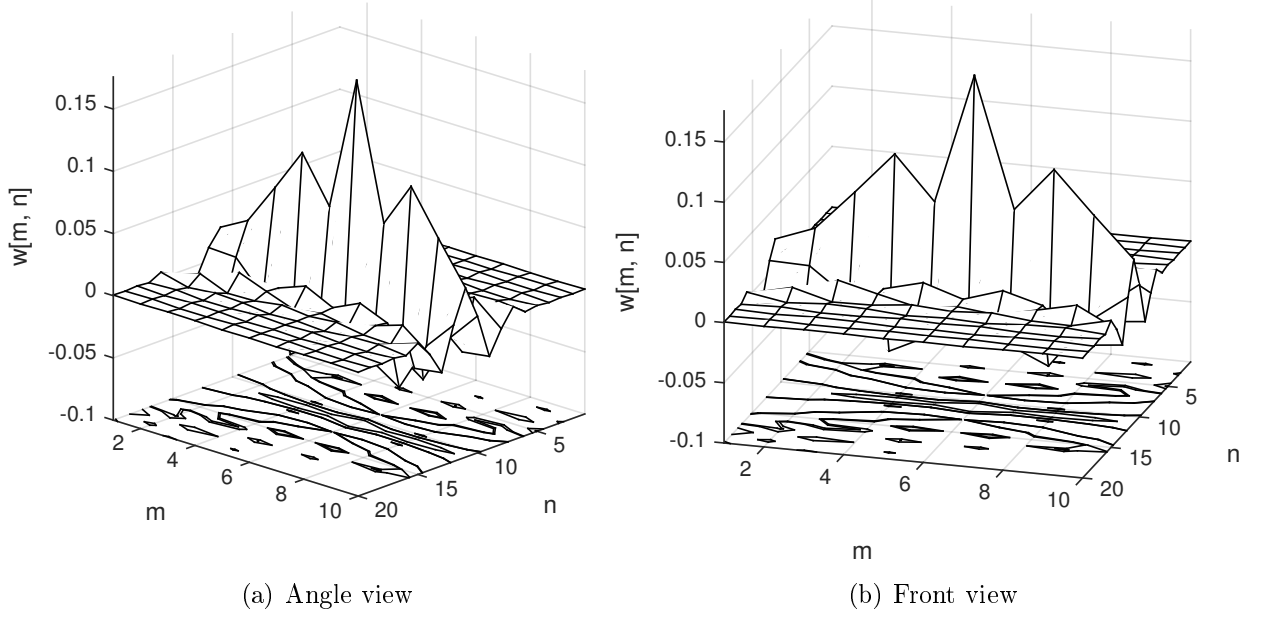


Figure 4.6: Taps of the filter $w[m, n]$

Figure 4.7 are considered.

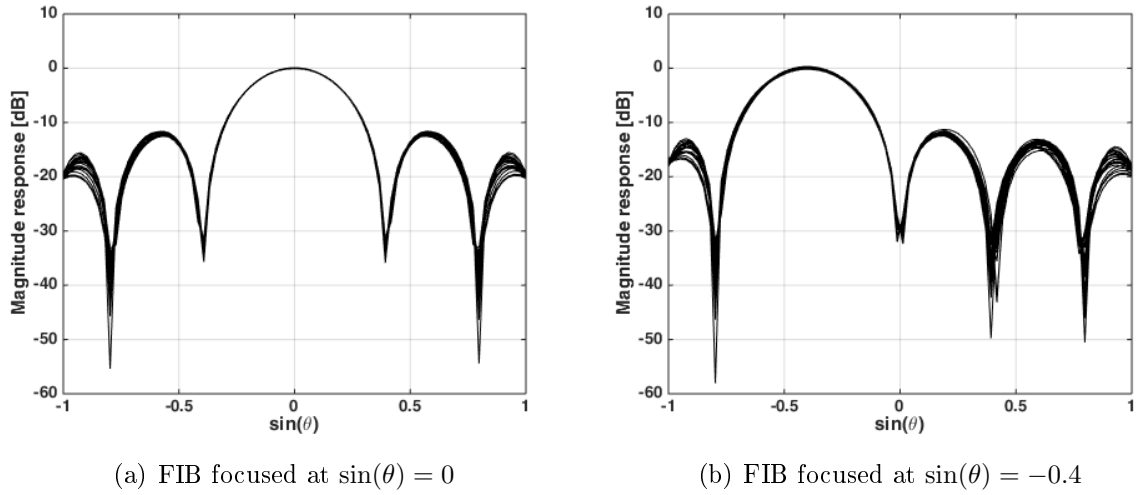


Figure 4.7: Responses of FIBs inside the frequency invariant region; (a) gain at $\sin(\theta) = -0.4$ is almost -35 dB and 0 dB at $\sin(\theta) = 0$. (b) the gain at $\sin(\theta) = 0$ is roughly -30 dB and 0 dB at $\sin(\theta) = -0.4$.

In Figure 4.7a, the lines show the frequency response over the invariant frequencies for a FIB focused at $\sin \theta = 0$, while Figure 4.7b shows the frequency response over the invariant frequencies for a FIB focused at $\sin \theta = 0.4$. When two signals impinging on the array from these exact same angles are considered and since the responses are practically frequency

invariant over the signal spectrum, the output from each FIB is an instantaneous sum of the source signals

$$x_{\text{fib}:0}(t) = P_{0,0}s_0(t) + P_{1,0}s_1(t), \quad (4.20)$$

$$x_{\text{fib}:1}(t) = P_{0,1}s_0(t) + P_{1,1}s_1(t), \quad (4.21)$$

where $x_{\text{fib}:k}(t)$ is the k -th FIB output signal. Generalizing for K FIBs, one can rewrite (4.20) and (4.21) in a compact matrix form

$$\mathbf{x}_{\text{fib}}(t) = \mathbf{P} \cdot \mathbf{s}(t) \in \mathbb{C}^{K \times 1}, \quad (4.22)$$

where \mathbf{P} is the matrix containing the gain coefficients $P_{i,k}$. This model is very similar to models one may find in narrowband scenarios such as the independent component analysis mixture model [50]. Therefore, narrowband algorithms may be applied with little or no modifications. Figure 4.8 shows a block diagram of a narrowband algorithm working via a bank of FIBs or the so called frequency invariance transformation.

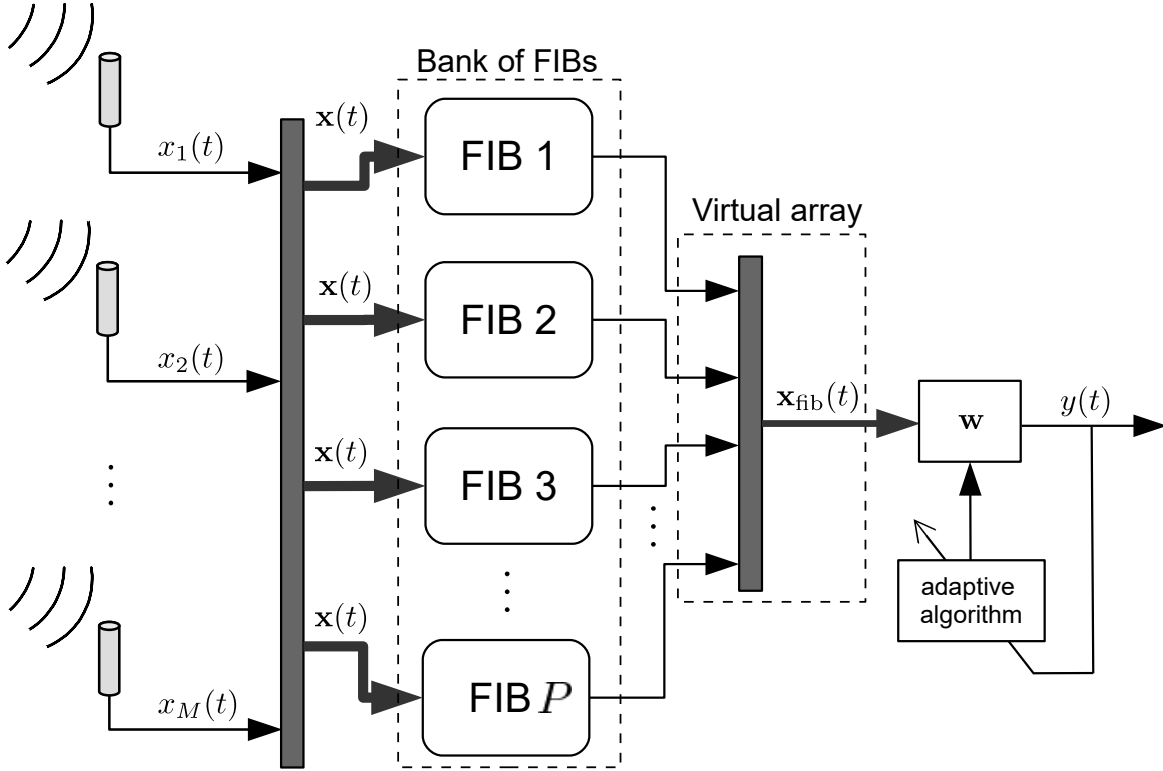


Figure 4.8: Diagram of a filtering scheme composed of a bank of FIBs and narrowband adaptive algorithm

According to [76], the maximum number of FIBs that can be used and still assure independence is $K_{\max} = \lfloor M/3 \rfloor$. Liu et. al. [65] developed a technique to shift the responses

appropriately to construct independent FIBs, avoiding the previous empirical method [76]. The concept of independence is important since an increased number of FIBs with dependent responses would cause an increase in computational complexity without interference cancellation gain. Liu et. al. [65] also showed that the maximum number of degrees of freedom of the prototype filter should be $M_{\text{prot}} = \min\{\lfloor M/3 \rfloor, \lfloor L/3 \rfloor\}$. Liu's method to provide independent responses consists on shifting the frequency axis of the k -th FIB by $\frac{2\pi k}{K}$, which can be seen as a multiplication in the frequency domain by the DFT coefficients $e^{-j\frac{2\pi k}{K}}$ of a DFT of size K . A way to accomplish the invariance is to shift the prototype filter (4.19) accordingly

$$F(\Omega_1/\Omega_2) = \frac{1}{M_{\text{prot}}} \sum_{m=-\lfloor M_{\text{prot}}/2 \rfloor}^{\lfloor M_{\text{prot}}/2 \rfloor} e^{-jm\pi(\sin\theta - \sin\theta_0)}, \quad (4.23)$$

where θ_0 is the direction to which the main beam is shifted. Using the DFT method, the k -th frequency invariant prototype filter is

$$F_k(\Omega_1/\Omega_2) = \frac{1}{M_{\text{prot}}} \sum_{m=-\lfloor M_{\text{prot}}/2 \rfloor}^{\lfloor M_{\text{prot}}/2 \rfloor} e^{-jm\pi\sin\theta} e^{-jm\frac{2\pi k}{K}}, \quad (4.24)$$

As an example, a bank of $K = 5$ FIBs is computed and the FIBs are plotted simultaneously in Figure 4.9 for a single frequency inside the invariance region.

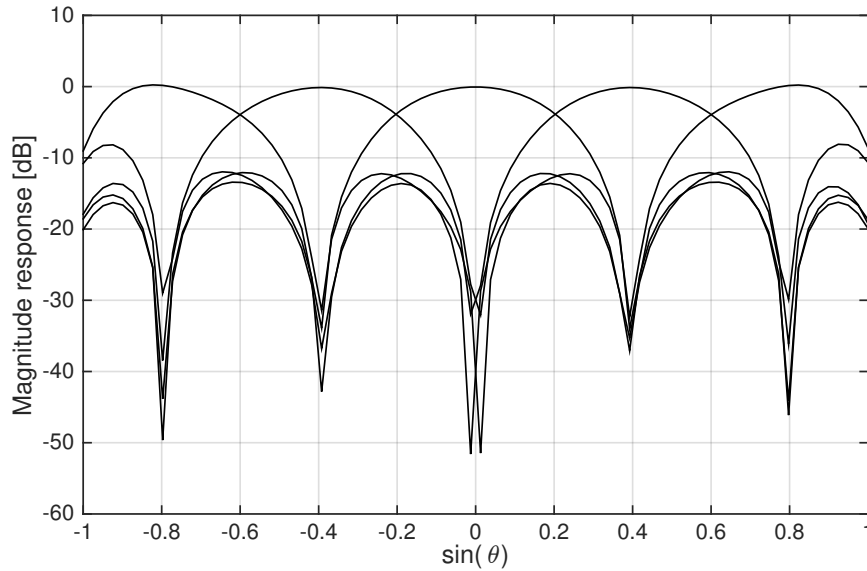


Figure 4.9: Response of a bank of 5 FIBs at a frequency inside the invariant range

4.4 Tensor Decomposition for Steering Matrix Estimation

In this section, the PARAFAC tensor model/decomposition is reviewed in Section 4.4.1 and a technique to estimate steering matrices using such model is revised in Section 4.4.2.

4.4.1 A Review on the PARAFAC Tensor Decomposition

Several tensor decompositions have been developed in the several years and have been revised, for example, in [54, 73]. One of the most used decompositions is the PARAFAC decomposition. One of the original works on this decomposition was published in 1970 [46] and also independently in the psychometrics field under the name canonical decomposition (CANDECOMP or CP) [21]. A 3-dimensional rank d PARAFAC decomposition is a sum of the outer products of vectors (rank-1 tensors) in all 3 dimensions

$$\mathcal{X} = \sum_{r=1}^d \mathbf{f}_r^{(1)} \circ \mathbf{f}_r^{(2)} \circ \mathbf{f}_r^{(3)}, \quad (4.25)$$

where $\mathbf{f}_r^{(l)} \in \mathbb{C}^{M_l \times 1}$. The process is also illustrated in Figure 4.10.

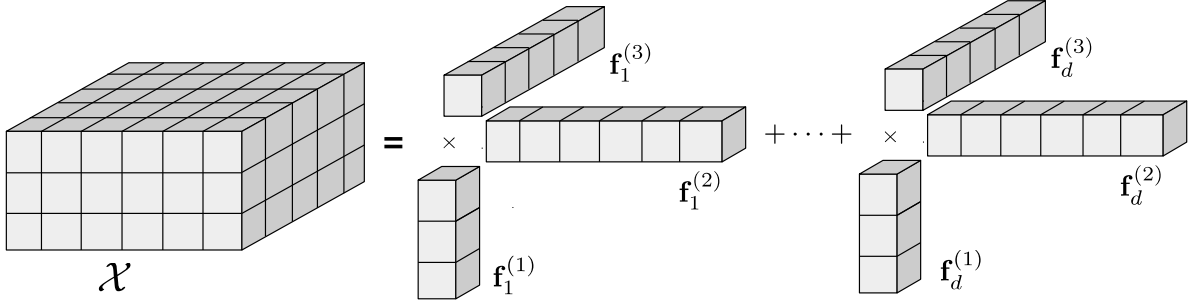


Figure 4.10: Illustration of a PARAFAC decomposition of a 3-dimensional (or three-way) array

It is also worth noting that the rank of the tensor is not directly related to the rank of a matrix, where its maximum rank is $d_{\max} = \min\{M_1, M_2\}$. For a tensor, only an upper bound is known which states that $d_{\max} \geq \min\{M_1, M_2, M_3\}$. In the case of array signal processing, this loose upper bound may allow for an underdetermined signal processing.

Since in the end matrices are what is usually estimated, another useful way to write the

PARAFAC decomposition is in a r -mode product notation [73]

$$\mathcal{X} = \mathcal{I}_3 \times_1 \mathbf{F}^{(1)} \times_2 \mathbf{F}^{(2)} \times_3 \mathbf{F}^{(3)} \in \mathbb{C}^{M_1 \times M_2 \times M_3}, \quad (4.26)$$

where \mathcal{I}_3 is a 3-dimensional identity tensor of size r and $\mathbf{F}^{(l)}$ are the factor matrices containing all r vectors in its columns. Another useful notation that allows for signal processing insights is a stacked matrices version

$$\mathcal{X} = [\mathbf{F}^{(1)} \cdot \text{diag}([\mathbf{F}^{(3)}]_{1,:}) \cdot \mathbf{F}^{(2)} \sqcup_3 \cdots \sqcup_3 \mathbf{F}^{(1)} \cdot \text{diag}([\mathbf{F}^{(3)}]_{M_3,:}), \cdot \mathbf{F}^{(2)}] \quad (4.27)$$

where $[\mathbf{F}^{(3)}]_{m,:}$ are the rows of $\mathbf{F}^{(3)}$ and \sqcup_3 denotes concatenation over the 3-rd dimension.

4.4.2 Alternate Least Squares for Steering Matrix Estimation

When trying to A very diffused approach to estimate the factor matrices of the PARAFAC decomposition is to use the alternate least squares (ALS) algorithm [54]. Its main idea is to write the D-dimensional tensor into D matrix forms (unfoldings) and apply the least squares solutions in each one of them. For a 3rd order tensor these matrices could be

$$[\mathcal{X}]_{(1)} = \mathbf{F}^{(1)} \cdot (\mathbf{F}^{(2)} \diamond \mathbf{F}^{(3)})^T \in \mathbb{C}^{M_1 \times M_2 \cdot M_3}, \quad (4.28)$$

$$[\mathcal{X}]_{(2)} = \mathbf{F}^{(2)} \cdot (\mathbf{F}^{(3)} \diamond \mathbf{F}^{(1)})^T \in \mathbb{C}^{M_2 \times M_3 \cdot M_1}, \quad (4.29)$$

$$[\mathcal{X}]_{(3)} = \mathbf{F}^{(3)} \cdot (\mathbf{F}^{(1)} \diamond \mathbf{F}^{(2)})^T \in \mathbb{C}^{M_3 \times M_1 \cdot M_2}, \quad (4.30)$$

where \diamond is the Khatri-Rao operator, i.e. a column-wise Kroenecker product.

Then, after a random initialization of the factors one can compute the least squares solution for each factor in an alternate manner:

$$\mathbf{F}^{(1)} = [\mathcal{X}]_{(1)} \cdot [(\mathbf{F}^{(2)} \diamond \mathbf{F}^{(3)})^T]^+, \quad (4.31)$$

$$\mathbf{F}^{(2)} = [\mathcal{X}]_{(2)} \cdot [(\mathbf{F}^{(3)} \diamond \mathbf{F}^{(1)})^T]^+, \quad (4.32)$$

$$\mathbf{F}^{(3)} = [\mathcal{X}]_{(3)} \cdot [(\mathbf{F}^{(1)} \diamond \mathbf{F}^{(2)})^T]^+. \quad (4.33)$$

Steps (4.31) to (4.33) run iteratively until a threshold condition ϵ is reached

$$\|\mathcal{X} - \mathcal{I}_3 \times_1 \mathbf{F}^{(1)} \times_2 \mathbf{F}^{(2)} \times_3 \mathbf{F}^{(3)}\|_{\text{H}}^2 \leq \epsilon, \quad (4.34)$$

where $\|\cdot\|_{\text{H}}^2$ is the higher order equivalent of the Frobenius norm and it is equal to the Frobenius norm of any of the unfoldings (4.28) - (4.30).

An alternative is to use transposed forward cyclical unfoldings [73] leading to the same

results

$$[\mathcal{X}]_{(1)}^{\odot T} = (\mathbf{F}^{(1)} \diamond \mathbf{F}^{(3)}) \cdot [\mathbf{F}^{(2)}]^T \in \mathbb{C}^{M_1 \cdot M_3 \times M_2}, \quad (4.35)$$

$$[\mathcal{X}]_{(2)}^{\odot T} = (\mathbf{F}^{(2)} \diamond \mathbf{F}^{(1)}) \cdot [\mathbf{F}^{(3)}]^T \in \mathbb{C}^{M_2 \cdot M_1 \times M_3}, \quad (4.36)$$

$$[\mathcal{X}]_{(3)}^{\odot T} = (\mathbf{F}^{(3)} \diamond \mathbf{F}^{(2)}) \cdot [\mathbf{F}^{(1)}]^T \in \mathbb{C}^{M_3 \cdot M_2 \times M_1}, \quad (4.37)$$

where, $[\cdot]_{(r)}^{\odot}$ is used to represent the forward cyclical form of the r -th mode unfolding. This form of unfolding is specially useful to derive adaptive tracking algorithms of the PARAFAC decomposition as explained in Section 4.5.

Now let us write the sampled time-delay correlation matrices of the noiseless data model (4.22)

$$\mathbf{R}(\tau) = \frac{1}{N} \mathbf{X}_{\text{fib}}(t) \mathbf{X}_{\text{fib}}^H(t - \tau) \in \mathbb{C}^{K \times K}, \quad (4.38)$$

$$\mathbf{R}(\tau) = \mathbf{P} \cdot \mathbf{R}_s(\tau) \cdot \mathbf{P}^H. \quad (4.39)$$

where $\mathbf{X}_{\text{fib}}(t) = [\mathbf{x}_{\text{fib}}(t) \ \mathbf{x}_{\text{fib}}(t - 1) \ \dots \ \mathbf{x}_{\text{fib}}(t - N + 1)]$ and $\mathbf{R}_s(\tau)$ is the sampled time-delay correlation matrix of the source signals. In the case that the source signals are non-white, i.e. signals are time correlated, set of J time-delay correlations can be written

$$\begin{cases} \mathbf{R}(\tau_1) &= \mathbf{P} \cdot \mathbf{R}_s(\tau_1) \cdot \mathbf{P}^H, \\ \mathbf{R}(\tau_2) &= \mathbf{P} \cdot \mathbf{R}_s(\tau_2) \cdot \mathbf{P}^H, \\ \vdots &\quad \quad \quad \vdots \\ \mathbf{R}(\tau_J) &= \mathbf{P} \cdot \mathbf{R}_s(\tau_J) \cdot \mathbf{P}^H. \end{cases} \quad (4.40)$$

By considering that $\mathbf{R}_s(\tau)$ is approximately diagonal, a direct link to the PARAFAC model via (4.27) can be made by considering $\mathbf{F}^{(1)} = \mathbf{P}$, $\mathbf{F}^{(2)} = \mathbf{P}^H$ and $\mathbf{F}^{(3)} = \mathbf{C}$, where $[\mathbf{C}]_{j,:} = \text{diag}(\mathbf{R}_s(\tau_j))$ [57]. Hence, Equation (4.40) can be written in tensor form as

$$\mathcal{R} = \mathcal{I} \times_1 \mathbf{P} \times_2 \mathbf{P}^H \times_3 \mathbf{C} \in \mathbb{C}^{K \times K \times J}. \quad (4.41)$$

In practice, a noise correlation term will be added to \mathcal{R} such that $\mathcal{R}_{\text{noisy}} = \mathcal{R} + \mathcal{V}$. However, if \mathcal{V} is sufficiently small, the ALS algorithm can be applied means of (4.31), (4.32) and (4.33) in order to find approximations of the factor matrices.

4.5 Proposed Adaptive Framework for Broadband Signals

The method proposed in this section is based on the PARAFAC recursive least squares tracking (RLST) algorithm proposed by Nion and Sidiropoulos [70]. The algorithm in [70] is suited for non-stationary narrowband signals. Therefore, a modification for the case where signals can be stationary yet having their non-whiteness explored is proposed. The result is a framework that applies the RLST with bank of FIBs for compatibility with broadband scenarios.

The basic idea of the RLST is to append a new slice to the correlation tensor for every newly acquired snapshot, as depicted in Figure 4.11. With a new snapshot, recursive update rules are applied in order to estimate the tensor factors. Also note that in Figure 4.11, the tensor is concatenated in the second dimension so that the forward cyclic notations can be used. Even though forward cyclic and reverse cyclic are equivalent, the forward cyclic unfoldings allow for a direct applications of the recursive least squares algorithms without the need to reshape the unfoldings.

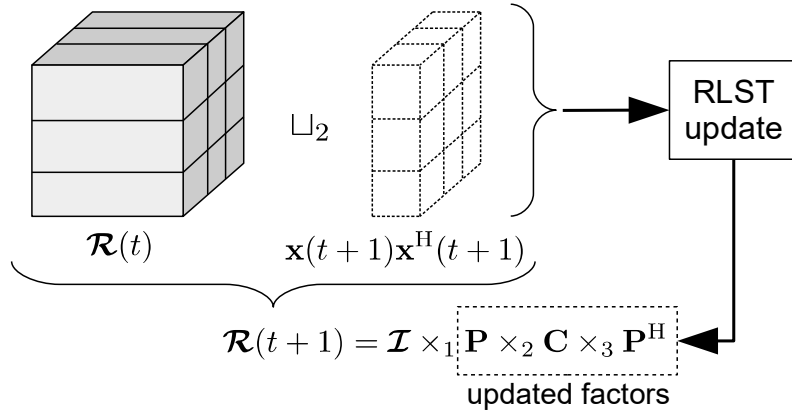


Figure 4.11: Concatenation of a new slice in the second dimension for a tensor with 3 dimensions

In [70], the correlation tensor grows with time, since the signals are considered are non-stationary and there is a different set of correlations for every time frame. In this work, the correlation tensor is practically stationary or slowly varying. However, for non-white signals, information from the time-delay correlations (4.42) can be extracted. The first problem arises from the broadband signal model, since the stack correlation matrices only contain information about the instantaneous mixtures. The second problem comes from the fact that the non-whiteness is only valid for a limited J time-delay and, therefore, the tensor no longer grows among one of the dimensions.

In order to solve the first problem above, a bank FIBs may be applied to the array output, as explained in Section 4.2, and a virtual array containing instantaneous mixtures $\mathbf{x}_{\text{fb}}(t)$ is obtained. For now, the input signal is generalized as an array signal $\mathbf{x}(t)$. For non-stationary signals, the PARAFAC RLST algorithm can be used directly by considering $\mathbf{x}_{\text{fb}}(t)$ as its signal input. From the second problem above, the PARAFAC RLST no longer applies, and a modification to the original algorithm output is needed.

To solve the second problem, the J time correlation matrices are updated via a weighted average (4.40) [83] for every new snapshot $\mathbf{x}(t+1)$

$$\begin{cases} \mathbf{R}(t+1, \tau_1) &= \lambda_r \mathbf{R}(t, \tau_1) + \mathbf{x}(t) \mathbf{x}^H(t - \tau_1), \\ \mathbf{R}(t+1, \tau_2) &= \lambda_r \mathbf{R}(t, \tau_2) + \mathbf{x}(t) \mathbf{x}^H(t - \tau_2), \\ \vdots & \vdots \\ \mathbf{R}(t+1, \tau_J) &= \lambda_r \mathbf{R}(t, \tau_J) + \mathbf{x}(t) \mathbf{x}^H(t - \tau_J), \end{cases} \quad (4.42)$$

where λ_r is a forgetting factor for the correlation matrices estimation. From (4.42), an updated tensor is obtained as whole, while [70] appends a new correlation matrix to the tensor. In order to still be able to use the PARAFAC RLST, a new correlation matrix is cyclically appended to the tensor and its older updated matrix is removed. The set of Equations (4.43) to (4.45) show an example on how the correlation tensor is updated. The same process is illustrated in Figure 4.12.

$$\mathcal{R}(t-1) = [\mathbf{R}(t-J, \tau_J) \sqcup_2 \mathbf{R}(t-J+1, \tau_1) \sqcup_2 \cdots \sqcup_2 \mathbf{R}(t-1, \tau_{J-1})], \quad (4.43)$$

$$\mathcal{R}(t) = [\mathbf{R}(t-J+1, \tau_1) \sqcup_2 \mathbf{R}(t-J+2, \tau_2) \sqcup_2 \cdots \sqcup_2 \mathbf{R}(t, \tau_J)], \quad (4.44)$$

$$\mathcal{R}(t+1) = [\mathbf{R}(t-J+2, \tau_2) \sqcup_2 \mathbf{R}(t-J+3, \tau_3) \sqcup_2 \cdots \sqcup_2 \mathbf{R}(t+1, \tau_1)], \quad (4.45)$$

where $\mathbf{R}(t-j, \tau_j)$ has size $K \times 1 \times K$. It is worth noting that the first time delay in (4.43) is not fixed to τ_J . It is only important that at the next snapshot, the delay shifts cyclically.

In the following, the PARAFAC RLST algorithm [70] is reviewed while using the cyclic updates as shown in (4.42) to (4.45) as input to the algorithm. First, the transposed forward cyclic 1st-mode unfolding of the tensor \mathcal{R} is taken

$$[\mathcal{R}(t)]_{(1)}^{\circ T} = (\mathbf{F}^{(1)}(t) \diamond \mathbf{F}^{(3)}(t)) [\mathbf{F}^{(2)}(t)]^T. \quad (4.46)$$

The factors $\mathbf{F}^{(1)}(t)$ and $\mathbf{F}^{(3)}(t)$ usually represent the channel matrices and should be slowly varying, hence, $\mathbf{H}(t) = \mathbf{F}^{(1)}(t) \diamond \mathbf{F}^{(3)}(t)$ may be used for further derivations. If the columns

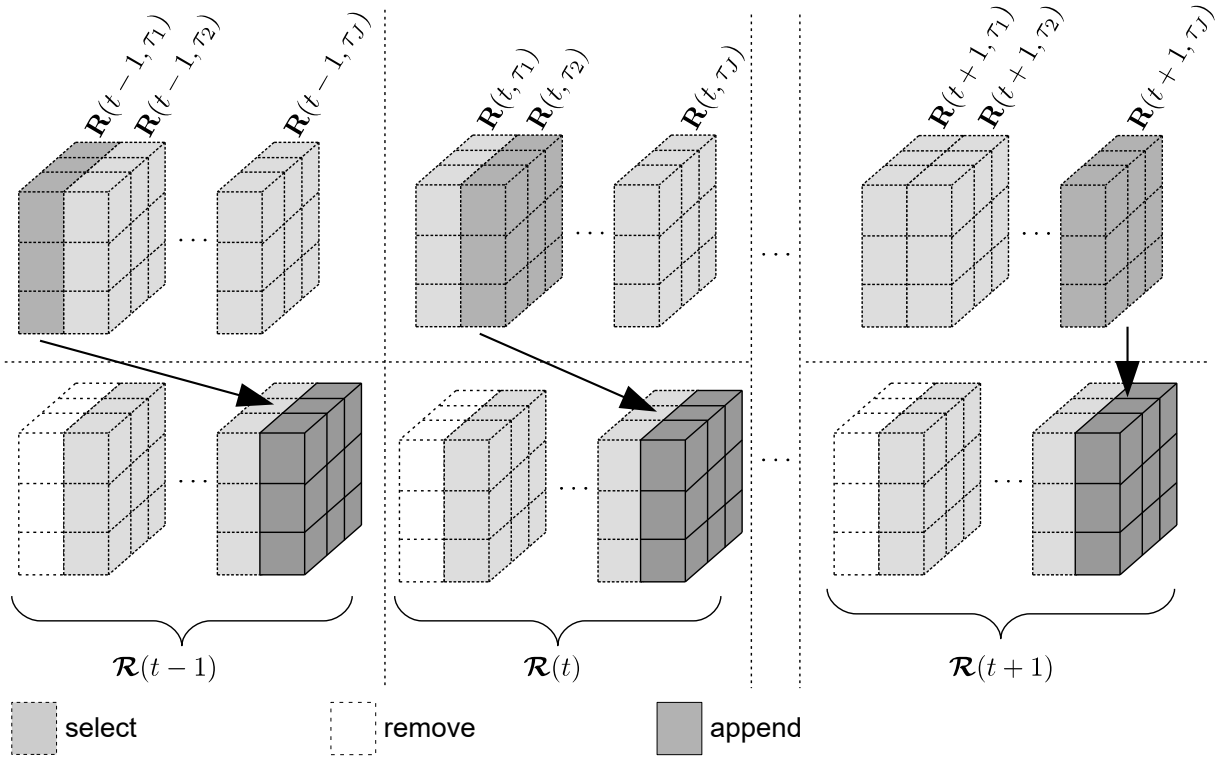


Figure 4.12: Process of selecting, removing and appending a new slice to the PARAFAC RLST input correlation tensor. After $\mathcal{R}(t+J-1)$ is updated, the process is restarted and the tensor is updated of with the new correlation matrix for the first time-delay τ_1 .

of $[\mathcal{R}(t)]_{(1)}^{\odot T}$ are analyzed, it is verified that

$$[\mathcal{R}(t+1)]_{(1)}^{\odot T} = \left[[[\mathcal{R}(t)]_{(1)}^{\odot T}]_{:, \bar{1}} \mathbf{r}(t+1) \right], \quad (4.47)$$

where $[\cdot]_{:, \bar{1}}$ corresponds to the operation of removing the first column of a matrix, $\mathbf{r}(t+1) = \text{vec}(\mathbf{R}(t+1, \tau_j))$ and τ_j is chosen cyclically from $\{\tau_1, \tau_2, \dots, \tau_J\}$. From the almost constant characteristics of $\mathbf{H}(t)$ it is found that

$$[\mathbf{F}^{(2)}(t+1)]^T \approx \left[[[\mathbf{F}^{(2)}(t)]^T]_{:, \bar{1}} [\mathbf{f}^{(2)}(t+1)]^T \right] \in \mathbb{C}^{J \times d}. \quad (4.48)$$

With knowledge of $\mathbf{H}(t)$, the new (transposed) line of the factor $[\mathbf{F}^{(2)}(t)]$ can be estimated using the least squares solution and, once $[\mathbf{f}^{(2)}(t+1)]^T$ is estimated, the new matrix $\mathbf{H}(t+1)$ can also be estimated via the least squares solution, thus

$$[\mathbf{f}^{(2)}(t+1)]^T = \mathbf{H}^+(t) \mathbf{r}(t+1), \quad (4.49)$$

$$\mathbf{H}(t+1) = [\mathcal{R}(t+1)]_{(1)}^{\odot T} [\mathbf{F}^{(2)}(t+1)]^{T+}. \quad (4.50)$$

In the next part, the focus is on finding a recursive the update for $\mathbf{H}(t)$. By writing the least squares cost function for a truncated window it is found that

$$\mathcal{J}_{\mathbf{H}, \mathbf{f}^{(2)}} = \sum_{j=1}^J \|\mathbf{r}(u+1) - \mathbf{H}(t+1)[\mathbf{f}^{(2)}(u+j)]^T\|_{\mathbf{F}}^2, \quad (4.51)$$

where $u = t + 1 - J$. In the original algorithm [70], a forgetting factor is also used. In this proposed framework case, only the newest correlations matrices are of interest, therefore, older estimates are vanished, which is equivalent to considering a rectangular window. In order to minimize the cost function (4.51) the gradient with respect to $\mathbf{H}(t+1)$ is taken and the result is solved for $\mathcal{J}_{\mathbf{H}, \mathbf{f}^{(2)}} \stackrel{!}{=} 0$

$$\nabla_{\mathbf{H}} \mathcal{J}_{\mathbf{H}, \mathbf{f}^{(2)}} = 2 \sum_{j=1}^J (\mathbf{r}(j) - \mathbf{H}(t+1)[\mathbf{f}^{(2)}(u+j)]^T) [\mathbf{f}^{(2)}(u+j)]^* \stackrel{!}{=} 0, \quad (4.52)$$

$$\mathbf{H}(t+1) = \left(\sum_{j=1}^J \mathbf{r}(u+j) [\mathbf{f}^{(2)}(u+j)]^* \right) \left(\sum_{j=1}^J [\mathbf{f}^{(2)}(u+j)]^T [\mathbf{f}^{(2)}(u+j)]^* \right)^{-1}. \quad (4.53)$$

The left side and right-side content inside parenthesis can be computed recursively as [70]

$$\mathbf{H}_p(t+1) = \mathbf{H}_p(t) + \mathbf{r}(t+1) [\mathbf{f}^{(2)}(t+1)]^* - \mathbf{r}(u) [\mathbf{f}^{(2)}(u)]^*, \quad (4.54)$$

$$\mathbf{H}_q(t+1) = \mathbf{H}_q(t) + [\mathbf{f}^{(2)}(t+1)]^T [\mathbf{f}^{(2)}(t+1)]^* - [\mathbf{f}^{(2)}(u)]^T [\mathbf{f}^{(2)}(u)]^*, \quad (4.55)$$

and update rule for $\mathbf{H}(t+1)$ can then be written as

$$\mathbf{H}(t+1) = \mathbf{H}_p(t+1) \mathbf{H}_q^{-1}(t+1). \quad (4.56)$$

In (4.56) the problem of high computational cost of a matrix inversion arises. However, from (4.56) two nested rank one matrix inversion lemmas can be written ¹. Once $\mathbf{H}(t)$ is updated, it is used to update $[\mathbf{f}^{(2)}(t)]^T$ by means of (4.49)

$$[\mathbf{f}^{(2)}(t+1)]^T = \mathbf{H}_q(t+1) \mathbf{H}_p^+(t+1) \mathbf{r}(t+1), \quad (4.59)$$

where an inversion problem arises one more time, this time, in the form of the pseudoinverse of $\mathbf{H}_p^+(t+1)$. From (4.54), one can conclude that a recursive update can be obtained by

¹ The matrix inversion lemma for a matrix $\mathbf{A} \in \mathbb{C}^{M \times M}$:

$$\mathbf{A}^{-1}(t+1) = (\mathbf{A}^{-1}(t) + \mathbf{b} \mathbf{c}^H)^{-1} \quad (4.57)$$

$$= \mathbf{A}^{-1}(t) + \frac{\mathbf{A}^{-1}(t) \mathbf{b} \mathbf{c}^H \mathbf{A}^{-1}(t)}{1 + \mathbf{c}^H \mathbf{A}^{-1}(t) \mathbf{b}} \quad (4.58)$$

applying the pseudoinverse lemma twice² Finally, the factors $\mathbf{F}^{(1)}(t+1)$ and $\mathbf{F}^{(3)}(t+1)$ have to be decomposed from $\mathbf{H}(t+1)$. By analyzing the Khatri-Rao product that forms $\mathbf{H}(t+1)$, it is concluded that its r -th column is $\mathbf{f}_r^{(1)}(t+1) \otimes \mathbf{f}_r^{(3)}(t+1) = \text{vec}(\mathbf{f}_r^{(1)}(t+1)[\mathbf{f}_r^{(3)}(t+1)]^T)$, where $\mathbf{f}_r^{(\cdot)}(t+1)$ is also the r -th column of $\mathbf{F}^{(\cdot)}(t+1)$. By undoing the vec operation, a rank-1 matrix $\mathbf{H}_r(t+1) = \text{unvec}([\mathbf{H}(t+1)]_{:,r})$ can be formed. Thus, a rank-1 SVD can be performed to separate $\mathbf{f}_r^{(1)}(t+1)$ and $\mathbf{f}_r^{(3)}(t+1)$ from $\mathbf{H}(t+1)$. This can be adaptively achieved by executing one iteration of the Bi-iterative SVD [79] as follows

$$\mathbf{f}_r^{(1)}(t+1) = \mathbf{H}_r^T(t+1)[\mathbf{f}_r^{(3)}(t)]^*, \quad (4.62)$$

$$\mathbf{f}_r^{(3)}(t+1) = \frac{\mathbf{H}_r(t+1)\mathbf{f}_r^{(1)}(t+1)}{\|\mathbf{H}_r(t+1)\|_F^2}. \quad (4.63)$$

Running the Bi-iterative SVD step along all $r = 1, \dots, d$, the final factor matrices $\mathbf{F}^{(1)}(t+1)$ and $\mathbf{F}^{(3)}(t+1)$ are estimated.

Making the correspondence with the previous tensor model (4.41), it is found that $\hat{\mathbf{P}} = \mathbf{F}^{(1)}, \mathbf{F}^{(2)} = \mathbf{C}$ and $\hat{\mathbf{P}}^H = \mathbf{F}^{(1)}$. Matrix \mathbf{P} contains in its rows information regarding each source. Since there is interference information in the remaining lines of $\hat{\mathbf{P}}$, a solution based on its pseudo-inverse where $\mathbf{W} = \hat{\mathbf{P}}^+$ can be found. However, due to the extra high complexity involved, it should be avoided. Unfortunately, the inversion of \mathbf{P} does not have a rank-1 update as previously occasions. The proposed solution is to apply one iteration of the Ben-Israel and Cohen [16] algorithm³ at each update step of the RLST algorithm, such that

$$\mathbf{W}(t+1) = \mathbf{W}(t)(2\mathbf{I}_K - \hat{\mathbf{P}}(t+1)\mathbf{W}(t)), \quad (4.67)$$

² The pseudo-inverse lemma for a matrix $\mathbf{A} \in \mathbb{C}^{M \times N}, M \geq N$ [18]:

$$\mathbf{A}^+(t+1) = (\mathbf{A}^+(t) + \mathbf{c}\mathbf{d}^H)^+ \quad (4.60)$$

$$= \mathbf{A}^+(t) + \frac{\mathbf{A}^+(t)\mathbf{h}^H\mathbf{u}^H}{\beta^*} - \frac{\beta^*\mathbf{p}\mathbf{q}^H}{\sigma}, \quad (4.61)$$

where $\beta = 1 + \mathbf{d}^H\mathbf{A}^+(t)\mathbf{c}$, $\mathbf{h} = \mathbf{d}^H\mathbf{A}^+$, $\mathbf{u} = (\mathbf{I}_M - \mathbf{A}(t)\mathbf{A}^+(t))\mathbf{c}$, $\mathbf{p} = -(\|\mathbf{u}\|^2/\beta^*\mathbf{A}^+(t)\mathbf{h}^H + \mathbf{k})$, $\mathbf{k} = \mathbf{A}^+(t)\mathbf{c}$, $\mathbf{q}^H = -(\|\mathbf{h}\|^2/\beta^*\mathbf{u}^H + \mathbf{h})$ and $\sigma = \|\mathbf{h}\|^2\|\mathbf{u}\|^2 + |\beta|^2$.

³ The Ben-Israel and Cohen iterative method [16] estimates the pseudo-inverse of $\mathbf{A} \in \mathbb{C}^{M \times N}$ after convergence of the following series

$$\hat{\mathbf{A}}^+(0) = \alpha\mathbf{A}^H \quad (4.64)$$

$$\hat{\mathbf{A}}^+(t+1) = \hat{\mathbf{A}}^+(t)(2\mathbf{I}_M - \mathbf{A}\hat{\mathbf{A}}^+(t)), \quad (4.65)$$

where α can be chosen inside the interval:

$$0 < \alpha < \frac{2}{\max_i \sum_j |[(\mathbf{A}^H\mathbf{A})]_{i,j}|}. \quad (4.66)$$

In the case an estimate of \mathbf{A}^+ is already available, it can be used to initialize the algorithm.

where \mathbf{I}_K is the identity matrix of size K and $\mathbf{W}(0)$ can be initialized with a good estimate of \mathbf{P}^+ , e.g. by taking the pseudo-inverse during the initialization of the PARAFAC RLST. To avoid non-convergence problems due to variations the input data, it may be useful to apply a time average over $\hat{\mathbf{P}}(t+1)$

$$\mathbf{W}(t+1) = \mathbf{W}(t)(2\mathbf{I}_K - \langle \hat{\mathbf{P}}(t+1) \rangle_t \mathbf{W}(t)), \quad (4.68)$$

where $\langle \cdot \rangle_t$ is a moving average such as $\langle \hat{\mathbf{P}}(t+1) \rangle_t = \lambda_t \hat{\mathbf{P}}(t+1) + (1 - \lambda_t) \hat{\mathbf{P}}(t)$ for $0 < \lambda_t \leq 1$. The above solution is denoted as RLST-Ben.

Alternatively, in order to remove interference in a direct and computationally efficient way, one can apply a narrowband adaptive beamformer where each row is a beamformer constraint. One of such algorithm is the Frost algorithm [35] with filter update

$$\mathbf{w}_i^{(\text{fr})}(t+1) = \frac{[\hat{\mathbf{P}}(t+1)]_{:,i}}{[\hat{\mathbf{P}}(t+1)]_{:,i}^H [\hat{\mathbf{P}}(t+1)]_{:,i}} + \mathbf{Q}_i (\mathbf{w}_i^{(\text{fr})}(t) - \mu_{\text{fr}} e^*(n) \mathbf{x}_{\text{fb}}(t)), \quad (4.69)$$

where $\mathbf{Q}_i = \mathbf{I}_k - [\hat{\mathbf{P}}(t+1)]_{:,i} ([\hat{\mathbf{P}}(t+1)]_{:,i}^H [\hat{\mathbf{P}}(t+1)]_{:,i})^{-1} [\hat{\mathbf{P}}(t+1)]_{:,i}^H$ and μ_{fr} is the step size. This method is denoted as the RLST-Frost. It is also worth noting that the time average $\langle [\hat{\mathbf{P}}(t+1)]_{:,i} \rangle_t$ is also important, since it allows enough steadiness of state so that the algorithm can converge.

In summary, this section proposed a new cyclical manner to construct the correlation tensor. This new tensor can be seen as a time growing tensor multiplied by a truncated rectangular window. This truncated tensor can be directly applied to the PARAFAC RLST adaptive algorithm to track the tensor decomposition. To find the final filter adaptively, Frost's algorithm and Ben-Israel and Cohen's algorithms where proposed as an extra step to the RLST algorithm. Table IV.1 summarizes the entire FIB-RLST-Ben and FIB-RLST-Frost process.

4.6 Simulations and Results

In this section, simulations for evaluation of the proposed scheme are conducted. First, set three sources impinging from 0° , 40° and -50° are set. Each signal is a QPSK signal containing 8 samples per symbol and filtered by raised cosine filter with a rolloff factor of 0.3. The signal of interest (SOI) is located at 0° and has its center frequency shifted to the normalized frequency $\Omega = 0.65\pi$. The interferences are centered at $\Omega = 0.75\pi$ and $\Omega = 0.85\pi$, respectively. The signals are received by a ULA of size $M = 17$ and the tap-delay line has length $L = 80$. A bank of FIBs is constructed for $P = 5$ virtual sensors. The

Table IV.1: Proposed FIB-RLST-Ben and FIB-RLST-Frost

Step	Description	Complexity
1. Initialization	a. Compute a bank of FIBs as described in Sec. 4.2-4.3 b. Find initial estimates for $\mathbf{F}^{(1)}$, $\mathbf{F}^{(2)}$ and $\mathbf{F}^{(3)}$, e.g. via ALS algorithm and $\mathbf{W} = [\mathbf{F}^{(1)}]^+$	
2. First update of $[\mathbf{f}^{(2)}]^T$	$[\mathbf{f}^{(2)}(t+1)]^T = \mathbf{H}_q(t)\mathbf{H}_p^+(t)\mathbf{r}(t+1)$	$8K^2(R+1)d$
3. Update \mathbf{H}_p	$\mathbf{H}_p(t+1) = \mathbf{H}_p(t) + \mathbf{r}(t+1)[\mathbf{f}^{(2)}(t+1)]^* - \mathbf{r}(u)[\mathbf{f}^{(2)}(u)]^*$	$14d^2$
4. Update \mathbf{H}_q	$\mathbf{H}_q(t+1) = \mathbf{H}_q(t) + [\mathbf{f}^{(2)}(t+1)]^T[\mathbf{f}^{(2)}(t+1)]^* - [\mathbf{f}^{(2)}(u)]^T[\mathbf{f}^{(2)}(u)]^*$	$14K^2d$
5. Update \mathbf{H}_p^{-1}	Apply the inverse matrix lemma twice to the result of Step 3 to find $\mathbf{H}_p^{-1}(t+1)$	$48d^2 + 20d$
6. Update \mathbf{H}_q^+	Apply the pseudoinverse matrix lemma twice to the result of Step 4 to find $\mathbf{H}_q^+(t+1)$	$96K^2d$
7. Update \mathbf{H}	$\mathbf{H}(t+1) = \mathbf{H}_p(t+1)\mathbf{H}_q^{-1}(t+1)$	$8K^2d^2$
8. Update $\mathbf{F}^{(1)}$ and $\mathbf{F}^{(3)}$	for $r = 1, \dots, d$ $\mathbf{f}_r^{(1)}(t+1) = \mathbf{H}_r^T(t+1)[\mathbf{f}_r^{(3)}(t)]^*$ $\mathbf{f}_r^{(3)}(t+1) = \frac{\mathbf{H}_r(t+1)\mathbf{f}_r^{(1)}(t+1)}{\ \mathbf{H}_r(t+1)\ _F^2}$	$8K^2d$ $8K^2d + 10Kd$
9. Second update of $[\mathbf{f}^{(2)}]^T$	$[\mathbf{f}^{(2)}(t+1)]^T = \mathbf{H}_q(t+1)\mathbf{H}_p^+(t+1)\mathbf{r}(t+1)$	$8d^2 + 8K^2d$
10. Update $\mathbf{F}^{(2)}$	$[\mathbf{F}^{(2)}(t+1)]^T = \begin{bmatrix} [\mathbf{F}^{(2)}(t)]^T \\ [\mathbf{f}^{(2)}(t+1)]^T \end{bmatrix}$	
11. Update \mathbf{W}	$\mathbf{W}(t+1) = 2\mathbf{W}(t) - \mathbf{W}(t)\mathbf{F}^{(2)}(t+1)\mathbf{W}(t)$ or $\mathbf{w}_i^{(\text{fr})}(t+1) = \frac{[\hat{\mathbf{P}}(t+1)]_{:,i}}{[\hat{\mathbf{P}}(t+1)]_{:,i}^H[\hat{\mathbf{P}}(t+1)]_{:,i}} + \mathbf{Q}_i(\mathbf{w}_i^{(\text{fr})}(t) - \mu_{\text{fr}}e^*(n)\mathbf{x}_{\text{fib}}(t))$	$2Kd + 16K^2d$ (*) $10K + 8K^2$ (**) $10dK + 8dK^2$ (*)

(*) for d sources, (**) for 1 source

used parameters are summarized in Table IV.2 and the resulting SINR averaged over 1000 simulations is shown in Figure 4.13.

Table IV.2: Parameters for the first scenario

Parameter	Value
M	17 antennas
L	80 taps
N	10000 snapshots
SNR	20 dB
P	5 virtual antennas
M_{prot}	P
λ_r	0.999
λ_t	0.98
ϵ	10^{-5}

In Figure 4.13, the batch PARAFAC estimated with ALS algorithm is used for comparison with the state of the art and upper bound. The FIB-RLST-Ben and the FIB-RLST-Frost are initialized at $N = 1000$ snapshots. The FIB-RLST-Ben tracks the batch PARAFAC

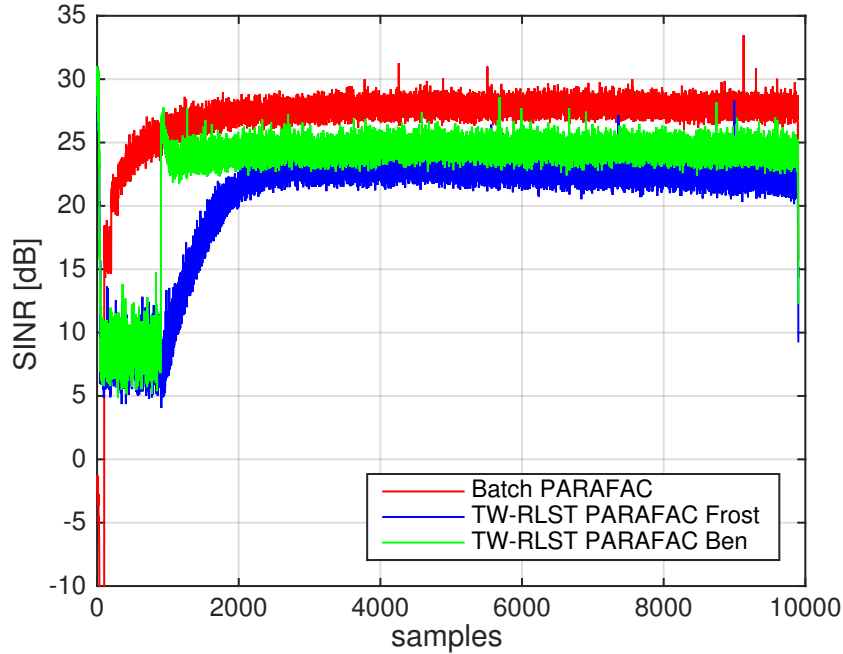


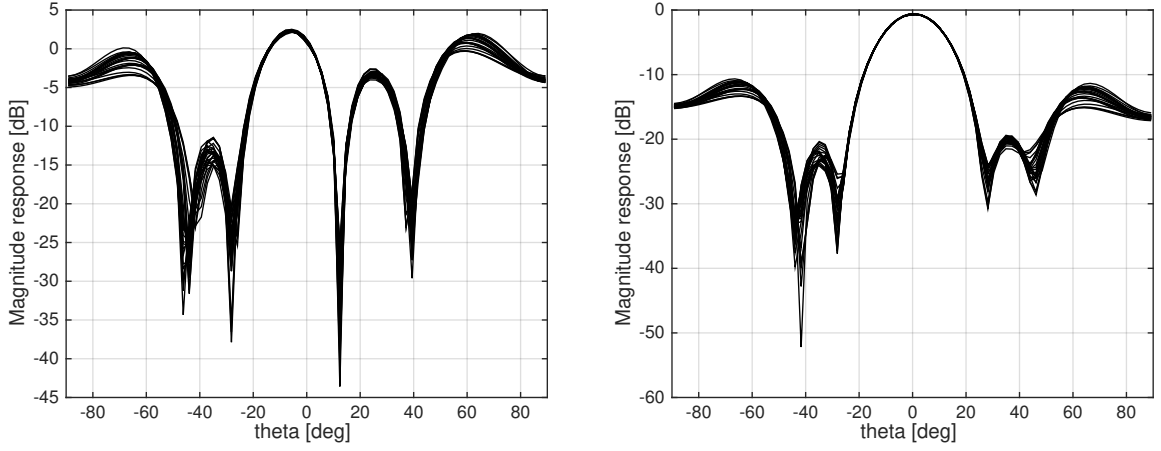
Figure 4.13: SINR for QPSK signals

very well with an SINR only roughly 2 dB bellow. The FIB-RLST-Frost still needs data input to reduce power from interference directions, however, its slower convergence may be compensated by its low computational complexity. In Figure 4.14, the beam responses show the interference cancellation at the respective direction for the FIB-RLST-Frost in Figure 4.14a and for the FIB-RLST-Ben in Figure 4.14b.

For the second set of simulations random complex Gaussian signals limited in band to the region $\Omega > 0.4\pi$ are generated. Each signal has a different spectrum changed by bandstop filter with cutoff frequencies at $[0.5 \ 0.6]\pi$, $[0.6 \ 0.7]\pi$ and $[0.7 \ 0.8]\pi$, respectively. The interferences start at a DOA of 30° and -55° and, at snapshot 5000, they are allowed to move 7° in a clockwise direction at the course of 100 snapshots. The obtained SINR after 1000 simulations is shown in Figure 4.15.

The resulting beam patterns are plotted in Figure 4.16 for the proposed FIB-RLST-Ben filter after $N = 5000$ snapshots and in 4.17 after $N = 10000$ snapshots.

For the final simulation the proposed method is compared with the method proposed in [63], which is based on the classic Gaussianity minimization approach [24]. For the simulation, two sources are set to impinge from 0° and 40° . The source at 0° is a BPSK signal containing 8 samples per symbol and filtered by the same raised cosine filter as in the first simulation. The SOI coming from 0° has its center frequency shifted to the normalized



(a) Spatial response for the FIB-RLST-Frost

(b) Beam response for the FIB-RLST-Ben

Figure 4.14: Spatial response over the invariant frequency band

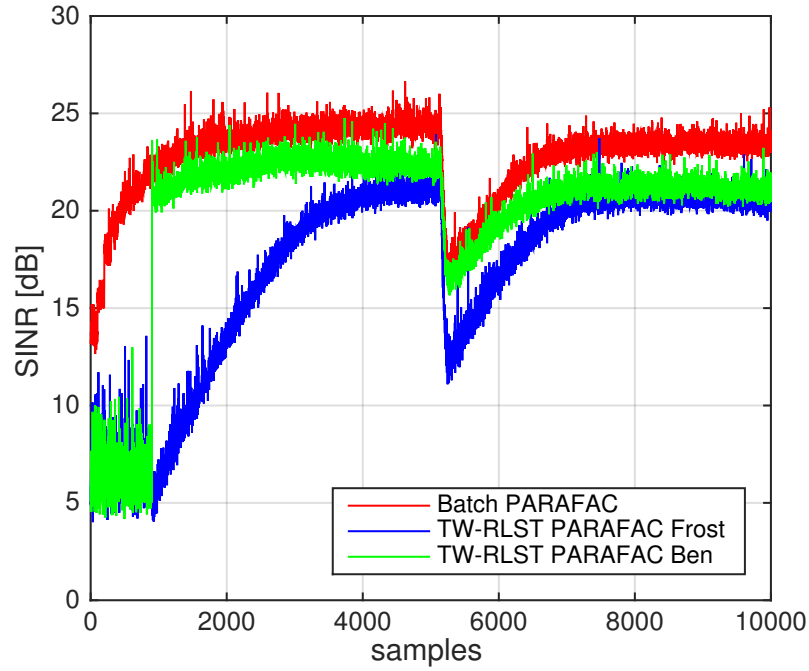
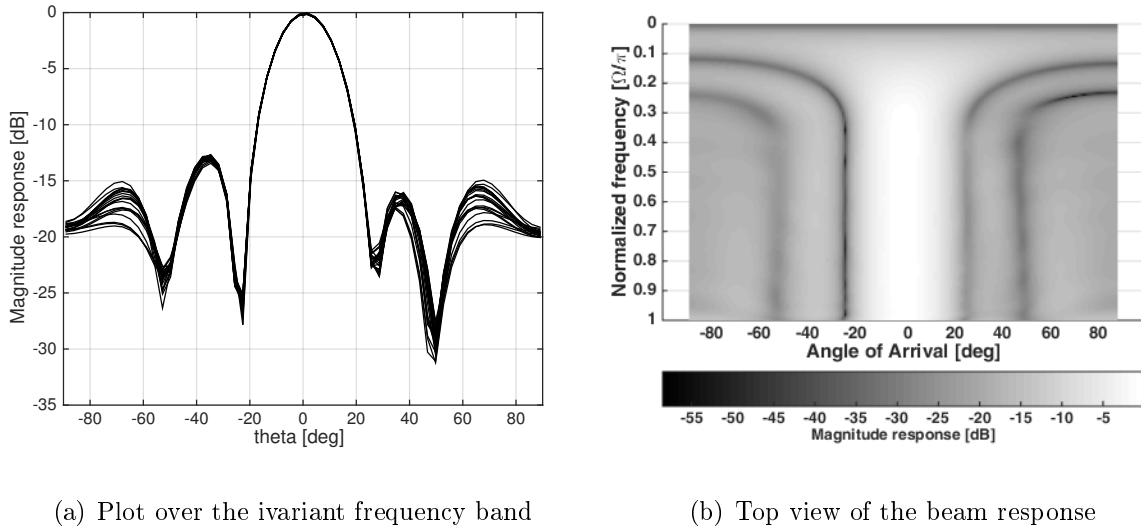
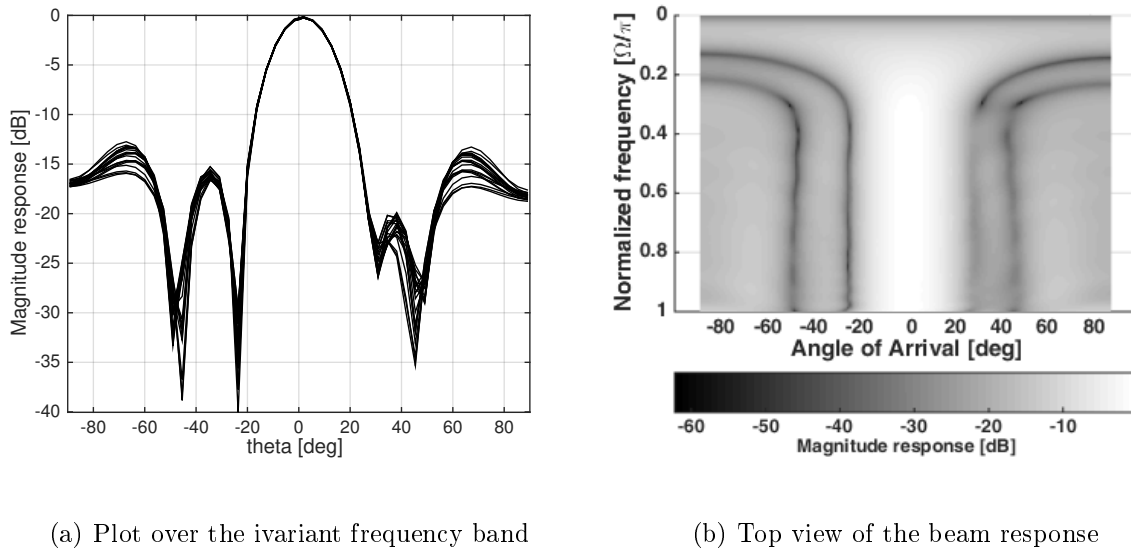


Figure 4.15: SINR for a varying DOA

frequency $\Omega = 0.65\pi$ and is also considered as the signal of interest (SOI). The second signal (interference) has a Laplacian distribution and its band is limited to the region $\Omega > 0.45\pi$, i.e. the invariance range of a FIB. The remaining parameters are configured as previously


 Figure 4.16: Beam response after $N = 5000$ of the proposed FIB-RLST-Ben filter

 Figure 4.17: Beam response after $N = 10000$ of the proposed FIB-RLST-Ben filter

shown in Table IV.2. After the raised cosine filtering operation, the SOI has a measured kurtosis of -0.65 and the interference has a measured kurtosis of +1.1 after having its band limited and the SNR is 20 dB. The output of the bank of FIBs is then processed by a kurtosis based ICA algorithm for comparison with the Batch ALS PARAFAC every 10 snapshots and the proposed adaptive schemes. The PARAFAC RLST algorithms are initialized after 800 snapshots. The SINR is computed and the result is shown in Figure 4.18.

In comparison to the ICA, the PARAFAC algorithm has a fast convergence. The idea of using an adaptive track of the PARAFAC algorithm is to take advantage of its fast conver-

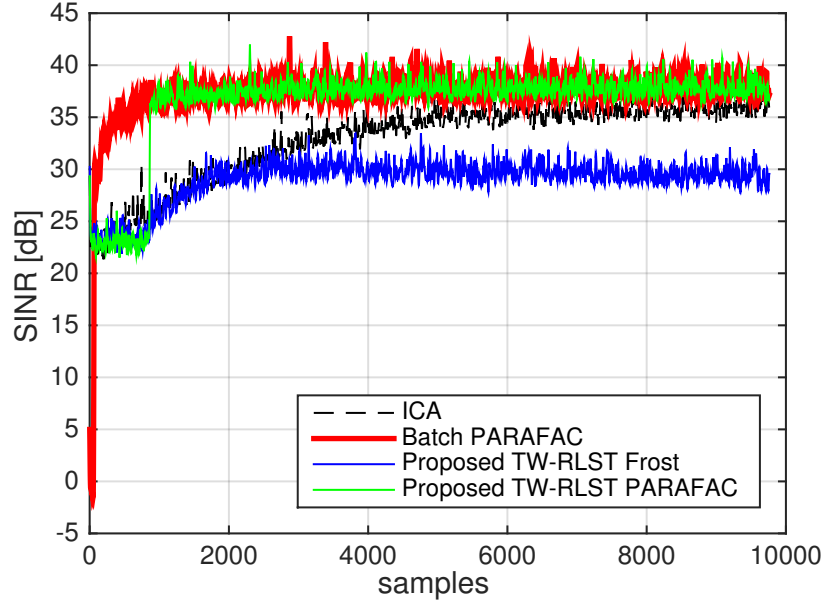
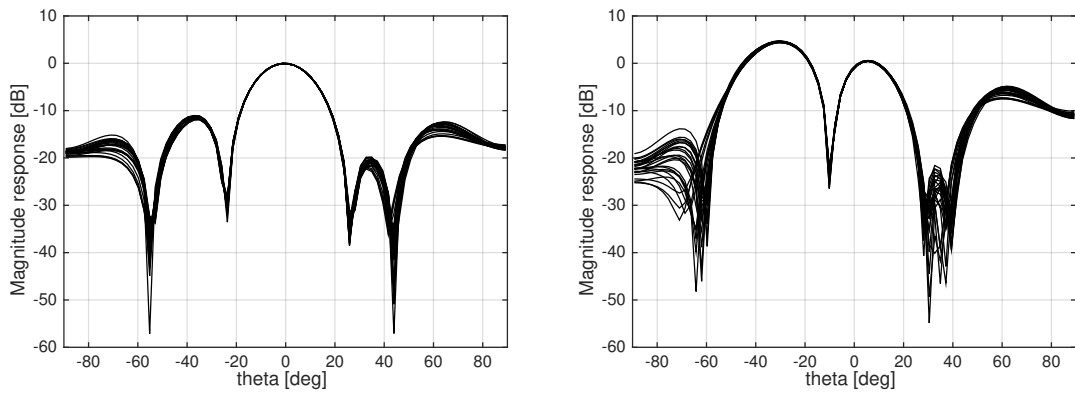


Figure 4.18: SINR comparison between ICA and the PARAFAC-based algorithms. The result is filtered by a moving average of length 10 for clarity.

gence and reduce its complexity to online processing levels. The proposed FIB-RLST-Ben pseudoinversion succeeds in tracking the PARAFAC decomposition and achieves practically the same SINR levels. The proposed FIB-RLST-Frost filtering showed a convergence similar to the ICA but achieves a final SINR roughly 8 dB below. Figure 4.19 shows the final beampatterns of the FIB-RLST-Ben and FIB-RLST-Frost algorithms with a clear null at the interference direction.



(a) Plot over the invariant frequency band of the FIB-RLST-Ben filter (b) Plot over the invariant frequency band of the FIB-RLST-Frost filter

Figure 4.19: Beampattern of the resulting FIB-RLST-Ben beamformer

4.7 Summary

Handling broadband signal may lead to high complexity algorithms. FIBs implement a frequency invariant transformation that transforms convolutive mixtures into instantaneous ones. The output signals have narrowband properties, which allow for the application of simpler techniques for separation of instantaneous mixtures. In this work, a framework for separating the signals that includes FIBs banks and the PARAFAC decomposition is proposed for adaptive scenarios. This adaptation is possible via a new method to construct the correlation tensor. Also, the framework includes two beamforming algorithms that are adapted over the PARAFAC decomposition output mixing matrices. The first algorithm has a low computational complexity, while the second has a high SINR that is comparable to batch algorithms. Finally, the proposed framework outperforms state-of-the-art ICA based approaches both in terms of SINR and in terms of faster convergence.

BEAMFORMER PERFORMANCE ASSESSMENT VIA THE UNSCENTED TRANSFORMATION

Applications such as speech and audio acquisition [3, 9, 48, 88], wireless communications [14] and RADAR [87] make use of array signal processing in order to enhance their capabilities. One of the most common uses of antenna arrays is spatial filtering by the use of beamformers [82]. However, idealistic assumptions such as a known direction of arrival (DOA) of the desired signal or perfectly spaced array elements are usually made [42]. Therefore, a performance assessment in the presence of deviations should be considered for practical implementations.

Geometry based beamformers, e.g. delay and sum (DS), generalized sidelobe cancellers (GSC) and minimum variance distortionless response (MVDR), take one or more DOAs as input parameters of the associated optimization problem. Though, DOA estimation is always prone to a certain degree of error. Moreover, the positioning of the antenna elements is not always perfectly known and it may affect the beamformer's quality. In this paper, the quality is measured as the average of the achieved signal to interference and noise ratio (SINR). The average is important, since the random nature of these perturbations will lead to random SINR values that may cause inconclusive results. For example, a simulation in a more favorable scenario may result in an SINR that is lower than that of a simulation in a less favorable one. However, these values vary around a mean and computing the average gives the system designer an overall SINR, i.e. which SINR level is expected when the system is subject to a certain degree of error.

The Monte Carlo (MC) method [41] is a commonly used simulation technique for the computation of the average SINR [62] due to its simplicity and easiness of implementation. However, it requires a large number of trials [17] to converge to a satisfactory result, implying

a long simulation time. Currently, performance assessment of embedded systems takes 30 % of the development time and it could increase to 70 % [86]. Therefore, improving the efficiency of performance assessment tools implies reducing production costs and delivering new solutions faster to the market.

Previous works derive analytical expressions to assess the system's quality by using first order expansions of the perturbed parameters [59, 74]. These analytical expression evaluate the perturbation due to noise and are exact for high signal to noise ratio (SNR) values, but they do not present a good fit for low SNR cases. In this paper, the effect of other type of perturbations are studied, more precisely the DOA estimation error and antenna element positioning error on the SINR. It is shown that, in these cases, the computation of such analytical expressions is hard or not practical and the use of the unscented transformation (UT) to numerically evaluate the SINR is proposed. The UT maps a continuous probability distribution into a discrete one with the same statistical moments [52]. When a non-linear function is applied to the mapped distribution, in this case the SINR function, the results give us a good fit in comparison to the traditional MC approach, yet with a negligible computational time for univariate perturbation models. When the perturbation is multivariate, e.g. error in the position of each antenna element, the complexity grows exponentially with the number of antennas. Therefore, the complexity is still lower than the MC method's complexity for a small number of antennas and greater for a large number of antennas. In order to alleviate the effect of the array spacing perturbation using the UT, the reader is referred to [67].

In this work, two perturbations are considered, a DOA estimation error and array element positioning error. We assume uncorrelated and equipowered sources so that the source signal covariance is an identity matrix and the covariance matrix can be computed as explained in Section 5.3. The evaluation of these perturbations gives raise to a univariate and a multivariate UT, respectively. For the sake of demonstration, one type of perturbation is considered for each case. In future work, the UT may also be applied for other types of perturbations such as frequency shift, mutual coupling, amplitude error and phase error. Also, other integration methods such as the quadrature and cubature transforms [15] showed better results than the UT for filtering purposes. Even though the quadrature and cubature transforms might also be considered for sensor array performance assessment, we regard them as future work and focus on the simplicity and ease of implementation of the UT.

The remainder of this paper is divided as follows. Section 5.1 shows the data model containing the considered perturbations. In Section 5.2, we overview basic concepts of the UT. In Section 5.3, we propose the performance assessment of a beamformer using the UT. Section 5.4 presents the simulations results. Finally, Section 5.5 draws the conclusions of this work.

5.1 Data Model

We start with the received signal model of an ideal uniform linear array (ULA) containing M antenna elements

$$\mathbf{x}(t) = \sum_i \mathbf{a}(\theta_i) s_i(t) + \mathbf{v}(t) \in \mathbb{C}^{M \times 1}, \quad (5.1)$$

where $\mathbf{a}(\theta_i)$ is the steering vector, θ_i is the DOA azimuth of i -th source signal $s_i(t)$ and $\mathbf{v}(t)$ is the additive noise term. Since we are dealing with a ULA, the steering vector contains the phase delays $a_{m,i} = e^{j2\pi\mu_i}$, where $\mu_i = (m-1)\frac{d}{\lambda} \cos \theta_i$, m is the antenna element index and d is the inter-element spacing in wavelengths. If d is given in wavelengths, the wavelength can be dropped from the phase delay μ_i .

Let us consider the case where DOA and element positioning errors are present as depicted in Figure 5.1.

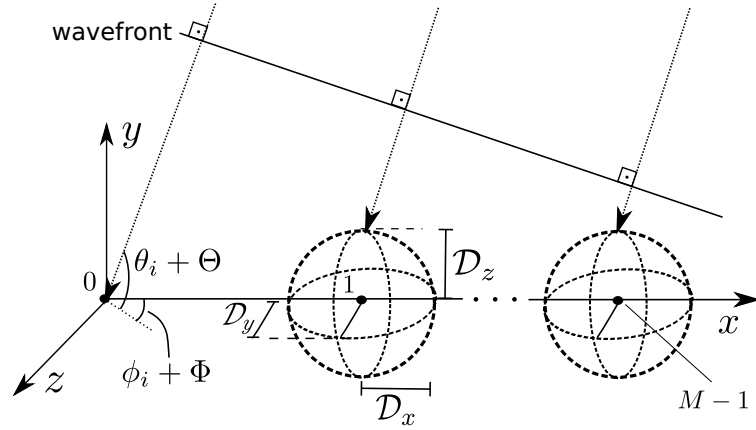


Figure 5.1: Illustration of a plane wavefront impinging on an antenna array containing DOA and element positioning errors

Figure 5.1 shows that a plane wavefront reaches the ULA with a DOA angle θ_i and is subject to an additive deviation represented by the Gaussian distributed random variable $\Theta \sim \mathcal{N}(0, \sigma_\theta^2)$. Therefore, the received signal in the presence of DOA estimation error becomes:

$$\mathbf{x}^{(\Theta)}(t) = \mathbf{a}^{(\Theta)}(\theta_0) s_0(t) + \sum_{i \neq 0} \mathbf{a}(\theta_i) s_i(t) + \mathbf{v}(t) \in \mathbb{C}^{M \times 1}, \quad (5.2)$$

where $\mathbf{a}^{(\Theta)}(\theta_0) = \mathbf{a}(\theta_0 + \Theta)$ is the steering vector perturbed by a random variable Θ and the signal of interest (SOI) corresponds to the $i = 0$ source signal. Note that (5.2) models the DOA estimation error as a physical change in the signal's direction.

Also, in Figure 5.1, the solid dots represent the antenna element positions. The first

element is the reference of the Cartesian axes. The remaining elements are positioned at $(m - 1)d$ along the x axis. Each of the elements, except the reference element, is subject to a positioning error in all of the 3 space dimensions x, y and z and are modeled by the random variables \mathcal{D}_x , \mathcal{D}_y and \mathcal{D}_z , respectively.

When the three dimensions are considered, not only the azimuth θ_i of the DOA, but also its elevation ϕ_i matters. Expanding the phase delays for the three dimensions we obtain

$$\begin{aligned} \mu_{m,i} = & (m - 1)d \cos \theta_i \cos \phi_i + \mathcal{D}_{x,m} \cos \theta_i \cos \phi_i \\ & + \mathcal{D}_{y,m} \sin \theta_i \cos \phi_i + \mathcal{D}_{z,m} \sin \phi_i, \end{aligned} \quad (5.3)$$

where the subscript m indicates the antenna index, since the random variables are independent with respect to each other.

The received signal for a random array positioning can be written as

$$\mathbf{x}^{(\mathcal{D})}(t) = \sum_i \mathbf{a}^{(\mathcal{D})}(\theta_i, \phi_i) s_i(t) + \mathbf{v}(t) \in \mathbb{C}^{M \times 1}, \quad (5.4)$$

where $\mathbf{a}^{(\mathcal{D})}(\theta_i, \phi_i) = [e^{j\mu_{1,i}}, e^{j\mu_{2,i}}, \dots, e^{j\mu_{M,i}}]^T$ is the steering vector perturbed by the random vector

$$\mathcal{D} = [\mathcal{D}_{x,1}, \mathcal{D}_{y,1}, \mathcal{D}_{z,1}, \mathcal{D}_{x,2}, \dots, \mathcal{D}_{z,M}].$$

5.2 Unscented Transformation

The Unscented Transformation (UT) is based on the mapping of a continuous probability distribution into a discrete one and can be used to compute the moments of non-linear transformations of a random variable [53]. Traditionally, such moments are computed via MC simulations. However, this approach requires large computational efforts and, depending on the accuracy, the computational complexity can be prohibitive.

In this section we review the concepts of the UT. The UT for a single random variable is reviewed in Section 5.2.1 and its extension for multiple i.i.d. random variables is reviewed in Section 5.2.2, respectively.

5.2.1 Univariate UT

Let us define the k -th moment of a continuous distribution [72]

$$\mu^{(\kappa)} = \mathbb{E}\{U^\kappa\} = \int_{-\infty}^{\infty} f_U(u) u^\kappa du, \quad (5.5)$$

where $f_U(u)$ is the probability density function (PDF) of the random variable U . Then, we make a discrete approximation of $f_U(\cdot)$,

$$f_U(u) \approx \omega(p_n) = \sum_n \omega_n \delta(u - p_n), \quad (5.6)$$

where p_n is the n -th UT sigma point, ω_n is the n -th weight and $\delta(u)|_{u=0} = 1$ and $\delta(u) = 0$ for any other values of u . Using (5.6), we write the discrete version of (5.5) [31]:

$$\hat{\mu}_{\text{ut}}^{(\kappa)} = \sum_n \omega_n p_n^\kappa, \quad (5.7)$$

To summarize, $\omega(p_n)$ is a discrete approximation of $f_U(u)$ with value (weight) ω_n at the sigma point p_n . The number of sigma points is chosen in accordance with the application and the desired accuracy. The weights and sigma points are calculated by solving the nonlinear system generated from the expansion of Equation (5.7) and by setting different values for κ , as follows:

$$\begin{cases} \omega_1 + \omega_2 + \dots = 1 \\ \omega_1 p_1 + \omega_2 p_2 + \dots = E\{U\} \\ \vdots \\ \omega_1 p_1^K + \omega_2 p_2^K + \dots = E\{U^K\}, \end{cases} \quad (5.8)$$

where K is the highest order used for the computations. The moments are known and the variable K depends on the number of sigma points and the type of distribution. Note that some moments might be zero depending on the distribution.

For instance, in the case of a Gaussian random variable with zero mean, we can take advantage of symmetry, i.e. $\omega_1 = \omega_3$, $p_1 = -p_3$ and $p_2 = 0$. Using this symmetry, we use 3 moments in order to find the 6 parameters resulting in $p_1 = -p_3 = \sqrt{3}\sigma^2$, $p_2 = 0$, $\omega_1 = \omega_3 = 1/6$ and $\omega_2 = 2/3$. Figure 5.2 compares a continuous PDF with its approximation using a 3-point UT and a 5-point UT. Note that the increase in UT points translates to a better PDF approximation such that the UT with infinite sigma points is equivalent to the continuous PDF.

The mapped points and weights hold the same moments as $f_U(\cdot)$ up to the biggest values used for κ . Therefore, we can replace the mean computed using MC simulation by the UT, i.e. the mean of a non-linear function of a random variable [31]

$$E\{g(U)\} = \sum_n \omega_n g(p_n), \quad (5.9)$$

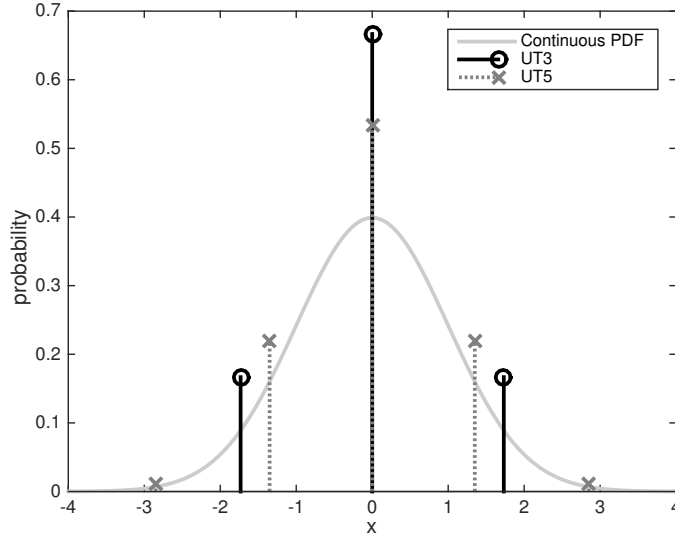


Figure 5.2: Comparison between a continuous PDF and a 3-point UT and 5-point UT PDF approximation

where $g(\cdot)$ is a non-linear function which depends on the application and scenario (in Section 5.3 we use the SINR function).

The fastest way to acknowledge Equation (5.9) is to expand $g(U)$ in its Taylor series

$$\begin{aligned} \mathbb{E}\{g(U)\} = & \mathbb{E}\left\{g(0) + \left.\frac{dg(u)}{du}\right|_{u=0} u + \frac{1}{2!} \cdot \left.\frac{d^2g(u)}{du^2}\right|_{u=0} u^2 \right. \\ & \left. + \frac{1}{3!} \cdot \left.\frac{d^3g(u)}{du^3}\right|_{u=0} u^3 + \dots \right\} \end{aligned} \quad (5.10)$$

$$= a_0 + a_1 \mathbb{E}\{U\} + a_2 \mathbb{E}\{U^2\} + a_3 \mathbb{E}\{U^3\} + \dots, \quad (5.11)$$

where a_i is the i -th Taylor series coefficient. Therefore, if the moments (5.5) and (5.7) are equal, Equation (5.9) is exact. However, by computing the UT, (5.11) is truncated at the K -th moment as defined in (5.8).

The condition on $g(\cdot)$ is that the derivatives should be continuous. Also note that the probability distribution of the original function is unknown and only its statistical moments are needed to compute the UT.

5.2.2 Multivariate UT

In the case where multiple random variables are present in the system model, a random vector $\mathbf{U} = [U_1, U_2, \dots, U_M]^T$ is considered. For i.i.d. random variables, the expected values of the cross-products are zero. Thus, the expected values of \mathbf{U} result in a set of

univariate integrals

$$\mathbb{E}[U_1^{\kappa_1}, \dots, U_m^{\kappa_m}, \dots, U_M^{\kappa_M}] = \int_{\mathbb{R}} f_{U_m}(u_m) u_m^{\kappa_m} du_m, \quad (5.12)$$

where $\kappa_m \in \{1, 2, \dots\}$ and $\kappa_{j \neq m} = 0$. Therefore, computing the UT for i.i.d. random variables simplifies to (5.8). Also, the expected value of $g(\mathbf{U}) : \mathbb{R}^M \rightarrow \mathbb{R}$ is expressed as

$$\begin{aligned} \mathbb{E}[g(\mathbf{U})] &= \int_{\mathbb{R}^M} f(u_1, u_2, \dots, u_M) g(\mathbf{u}) d\mathbf{u} \\ &= \int_{\mathbb{R}^M} \prod_m f_{U_m}(u_m) g(\mathbf{u}) d\mathbf{u}. \end{aligned} \quad (5.13)$$

Since the multivariate polynomial can be written as the product of univariate polynomials [32] and by considering that $g(\mathbf{U})$ is approximated by a multivariate polynomial, Equation (5.13) can be easily computed if \mathbf{U} is i.i.d. Hence, (5.13) becomes a product of univariate integrals leading to the final result in Equation (5.18). However, the polynomial approximation can be dropped by using (5.6) and the i.i.d. assumption to get the multivariate PDF

$$f_{\mathbf{U}}(\mathbf{u}) \approx \prod_m \sum_{n_m} \omega_{n_m} \delta(u_m - p_{n_m}). \quad (5.14)$$

Via (5.14), $\mathbb{E}[g(\mathbf{U})]$ can be evaluated without the polynomial assumption in the following way

$$\mathbb{E}[g(\mathbf{U})] \approx \int_{\mathbb{R}^M} g(\mathbf{u}) \prod_m \sum_{n_m} \omega_{n_m} \delta(u_m - p_{n_m}) d\mathbf{u} \quad (5.15)$$

$$= \int_{\mathbb{R}^M} g(\mathbf{u}) \sum_{n_1} \cdots \sum_{n_M} \prod_m \omega_{n_m} \delta(u_m - p_{n_m}) d\mathbf{u} \quad (5.16)$$

$$= \sum_{n_1} \cdots \sum_{n_M} \int_{\mathbb{R}^M} g(\mathbf{u}) \prod_m \omega_{n_m} \delta(u_m - p_{n_m}) d\mathbf{u}. \quad (5.17)$$

Using the multidimensional sifting property in (5.17), the UT mapping is found for the i.i.d. multiple random variables

$$\mathbb{E}[g(\mathbf{U})] \approx \sum_{n_1} \cdots \sum_{n_M} g(\mathbf{p}_{n_1, \dots, n_M}) \left(\prod_m \omega_{n_m} \right), \quad (5.18)$$

where $\mathbf{p}_{n_1, \dots, n_M} = [p_{n_1}, \dots, p_{n_M}]^T$ is the vector of sigma points.

5.3 Proposed Performance Assessment of Array Response via UT

In this section, a beamforming performance evaluation technique in the presence of DOA and positioning deviations using the UT is proposed. For the beamformer, the classical Delay and Sum (DS) technique [37] is chosen. The DS filter is $\mathbf{w} = \mathbf{a}(\theta_0)$ and the goal is to evaluate the signal to interference plus noise ratio (SINR). The SINR for uncorrelated signals is

$$\text{SINR} = \frac{\mathbf{w}^H \mathbf{R}_{\text{ss}} \mathbf{w}}{\mathbf{w}^H (\mathbf{R}_{\text{int}} + \sigma^2 \mathbf{I}) \mathbf{w}}, \quad (5.19)$$

where σ^2 is the white noise variance. For uncorrelated signals, the SOI correlation matrix \mathbf{R}_{ss} and the interference correlation matrix \mathbf{R}_{int} become

$$\mathbf{R}_{\text{ss}} = \mathbf{a}(\theta_0) \mathbf{a}^H(\theta_0), \quad (5.20)$$

$$\mathbf{R}_{\text{int}} = \sum_{i \neq 0} \mathbf{a}(\theta_i) \mathbf{a}^H(\theta_i). \quad (5.21)$$

The result of inserting perturbations, as modeled in (5.2) and (5.4), implies the replacement of $\mathbf{a}(\theta_i)$ for $\mathbf{a}^{(\Theta)}(\theta_i)$ in (5.20) and (5.21) for the case where DOA estimation error is considered. When perturbations on the antenna element positions are considered, $\mathbf{a}(\theta_i)$ is replaced by $\mathbf{a}^{(\mathcal{D})}(\theta_i)$. From the random nature of $\mathbf{a}^{(\Theta)}(\theta_i)$ and $\mathbf{a}^{(\mathcal{D})}(\theta_i)$, it becomes clear that the SINR is also random. However, it is expected that the random SINR values vary around a mean. The mean can be used as a reference by the system designer to verify the impact of the DOA estimation error and element positioning error on the output SINR. In Sections 5.3.1 and 5.3.2, the classical MC approach and the proposed UT approach are presented for the computation of the SINR mean. In this paper, the averaged SINR will be denoted as $\overline{\text{SINR}}$.

5.3.1 Univariate case: Evaluating the SINR under DOA estimation error

First, the univariate case is considered, i.e. a DOA estimation error is present. This error is univariate because the same perturbation equally affects all the elements of $\mathbf{a}_{\Theta}(\theta_0 + \Theta)$. When the random variable is independent for each element of the steering vector, it is said to be multivariate (see Section 5.3.2).

The traditional method for evaluating the SINR is to compute the mean of the SINR over

N realizations of Θ . The average value is then expressed as:

$$\overline{\text{SINR}}_{\text{MC}}^{(\Theta)} = \frac{1}{N} \sum_{n=1}^N \frac{\mathbf{w}^H \mathbf{R}_{\text{ss}}^{(\Theta)}(\Theta[n]) \mathbf{w}}{\mathbf{w}^H (\mathbf{R}_{\text{int}}^{(\Theta)}(\Theta[n]) + \sigma^2 \mathbf{I}) \mathbf{w}}, \quad (5.22)$$

where $\Theta[n]$ is the n -th realization of Θ , and the resulting correlation matrices are given by:

$$\mathbf{R}_{\text{ss}}^{(\Theta)}(\Theta[n]) = \mathbf{a}(\theta_0 + \Theta[n]) \mathbf{a}^H(\theta_0 + \Theta[n]), \quad (5.23)$$

$$\mathbf{R}_{\text{int}}^{(\Theta)}(\Theta[n]) = \sum_{i \neq 0} \mathbf{a}(\theta_i + \Theta[n]) \mathbf{a}^H(\theta_i + \Theta[n]). \quad (5.24)$$

Note that this average can be very time consuming since it could take thousands of realizations to average the SINR to a reasonable level. In order to reduce the computational burden of such calculations, the use of (5.9) to average the SINR in a more computationally efficient way is proposed. Typically, less than 10 samples need to be generated, reducing drastically the computational time in comparison to MC simulations. With the knowledge of a few first moments of the random variable Θ , the UT weights and sigma points are deterministically computed as described in Section 5.2. The UT average of the SINR is obtained by replacing $g(\cdot)$ in Equation (5.9) by the SINR and Θ_n by $p_n^{(\Theta)}$:

$$\overline{\text{SINR}}_{\text{UT}}^{(\Theta)} = \sum_{n=1}^{N_{\text{UT}}} \omega_n \frac{\mathbf{w}^H \mathbf{R}_{\text{ss}}^{(\Theta)}(p_n^{(\Theta)}) \mathbf{w}}{\mathbf{w}^H (\mathbf{R}_{\text{int}}^{(\Theta)}(p_n^{(\Theta)}) + \sigma^2 \mathbf{I}) \mathbf{w}}. \quad (5.25)$$

Since typically $N_{\text{UT}} \ll N_{\text{MC}}$, Equation (5.25) turns out to be an efficient way to compute the SINR average. Figure 5.3 shows the comparison between N_{UT} and N_{MC} for the 3 and 5-point UT and N_{MC} ranging from 10^2 to 10^5 realizations.

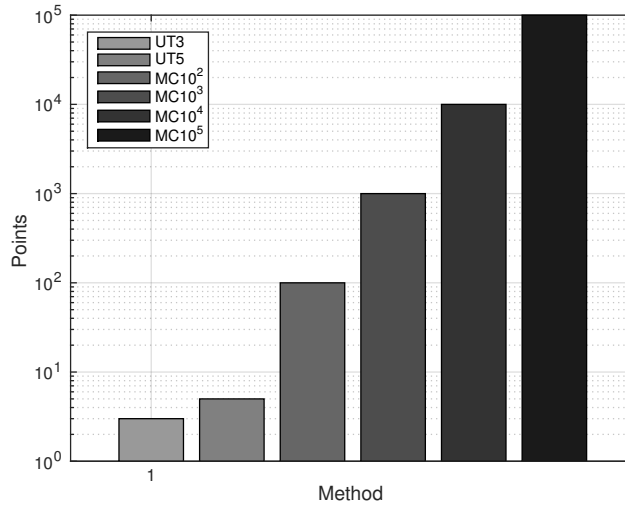


Figure 5.3: Number of generated random samples

5.3.2 Multivariate case: Evaluating the SINR under antenna element positioning error

In this Section, the case where a beamforming system is perturbed by an antenna element positioning error is considered. Since the perturbation occurs independently in each element, a multivariate case needs to be taken into consideration. The traditional Monte Carlo method consists of taking the mean over N realizations of the random vector \mathcal{D} yielding

$$\overline{\text{SINR}}_{\text{MC}}^{(\mathcal{D})} = \frac{1}{N} \sum_n \frac{\mathbf{w}^H \mathbf{R}_{\text{ss}}^{(\mathcal{D})}(\mathcal{D}[n]) \mathbf{w}}{\mathbf{w}^H (\mathbf{R}_{\text{int}}^{(\mathcal{D})}(\mathcal{D}[n]) + \sigma^2 \mathbf{I}) \mathbf{w}}. \quad (5.26)$$

In a similar manner as in (5.23) and (5.24), the SOI and interference correlation matrices in the presence of element positioning perturbations are written as:

$$\mathbf{R}_{\text{ss}}^{(\mathcal{D})}(\mathcal{D}[n]) = \mathbf{a}^{(\mathcal{D}[n])}(\theta_0, \phi_0) [\mathbf{a}^{(\mathcal{D}[n])}(\theta_0, \phi_0)]^H, \quad (5.27)$$

$$\mathbf{R}_{\text{int}}^{(\mathcal{D})}(\mathcal{D}[n]) = \sum_{i \neq 0} \mathbf{a}^{(\mathcal{D}[n])}(\theta_i, \phi_i) [\mathbf{a}^{(\mathcal{D}[n])}(\theta_i, \phi_i)]^H. \quad (5.28)$$

By analyzing Equation (5.18) to find a UT equivalent for (5.26), it is seen that a sum over all random variables is needed. Therefore, the result is a sum over the three dimension for each antenna element

$$\begin{aligned} \overline{\text{SINR}}_{\text{UT}}^{(\mathcal{D})} &= \sum_{n_{x,1}=1}^{N_{\text{UT}}} \sum_{n_{y,1}=1}^{N_{\text{UT}}} \sum_{n_{z,1}=1}^{N_{\text{UT}}} \sum_{n_{x,2}=1}^{N_{\text{UT}}} \dots \\ &\quad \sum_{n_{z,M}=1}^{N_{\text{UT}}} \omega_{n_{x,1}} \omega_{n_{y,1}} \omega_{n_{z,1}} \omega_{n_{x,2}} \dots \omega_{n_{z,M}} \cdot \\ &\quad \cdot \frac{\mathbf{w}^H \mathbf{R}_{\text{ss}}^{(\mathcal{D})}(\mathbf{p}_{n_{x,1}, \dots, n_{z,M}}) \mathbf{w}}{\mathbf{w}^H (\mathbf{R}_{\text{int}}^{(\mathcal{D})}(\mathbf{p}_{n_{x,1}, \dots, n_{z,M}}) + \sigma^2 \mathbf{I}) \mathbf{w}}, \end{aligned} \quad (5.29)$$

where $\mathbf{p}_{n_{x,1}, \dots, n_{z,M}} = [p_{n_{x,1}}, p_{n_{y,1}}, p_{n_{z,1}}, p_{n_{x,2}}, \dots, p_{n_{z,M}}]^T$.

Even though Gaussian distributions have non-zero tails, the actual generation of very large or very small numbers is unlikely. This work, as in common practice [34, 58], uses Gaussian distributions to model the array elements positioning errors in order to simplify the UT validation while making very little to no compromise to the validation process. For a theoretically more precise distribution, the reader is referred to the circular normal distribution, also known as the Von Mises distribution [49]. The characteristic of a Gaussian distribution allows us to simplify (5.29) by considering \mathcal{D}_x , \mathcal{D}_y and \mathcal{D}_z as zero mean Gaussian random variables with variance σ_d^2 . Since the sum of Gaussian random variables results in a Gaussian distribution with variance equal to the sum of the original variances [72], the

simplified perturbed phase delays are considered

$$\mu_{m,i}^{(\mathcal{D})} = (m-1)d \cos \theta_i \cos \phi_i + \mathcal{D}_m, \quad (5.30)$$

where $\mathcal{D}_m \sim \mathcal{N}(0, \sigma_{\mathcal{D}}^2)$. The total variance $\sigma_{\mathcal{D}}^2 = \sigma_d^2 (\cos^2 \theta_i \cos^2 \phi_i + \sin^2 \theta_i \cos^2 \phi_i + \sin^2 \phi_i)$ is actually equal to σ_d^2 , given that the sum inside the parenthesis is equal to 1 after the use of trigonometric identities. Therefore, the mean SINR computed via the UT for a perturbed element positioning can be simplified to

$$\begin{aligned} \overline{\text{SINR}}_{\text{UT}}^{(\mathcal{D})} &= \sum_{n_1=1}^{N_{\text{UT}}} \sum_{n_2=1}^{N_{\text{UT}}} \cdots \sum_{n_M=1}^{N_{\text{UT}}} \omega_{n_1} \omega_{n_2} \cdots \omega_{n_M} \\ &\cdot \frac{\mathbf{w}^H \mathbf{R}_{\text{ss}}^{(\mathcal{D})}(\mathbf{p}_{n_1, n_2, \dots, n_M}) \mathbf{w}}{\mathbf{w}^H (\mathbf{R}_{\text{int}}^{(\mathcal{D})}(\mathbf{p}_{n_1, n_2, \dots, n_M}) + \sigma^2 \mathbf{I}) \mathbf{w}}. \end{aligned} \quad (5.31)$$

In the multivariate case, the amount of points to be averaged by the UT varies not only with the amount of sigma points, but also with the amount of antenna elements. The comparison between the number of points needed for the MC and UT averages are shown in Figure 5.4.

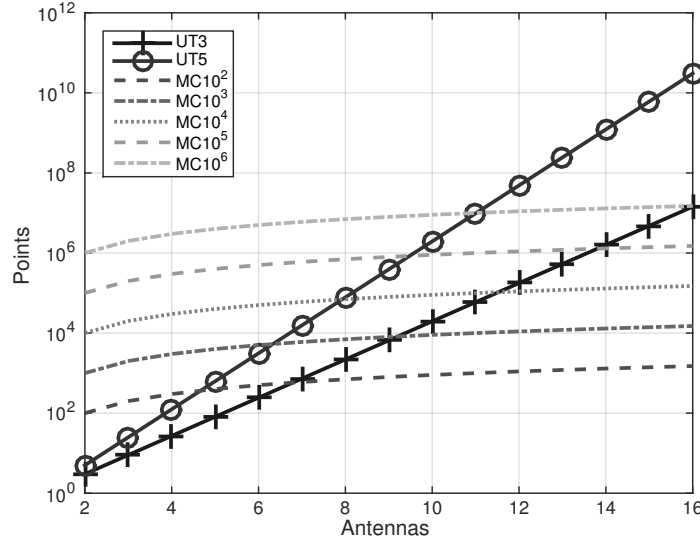


Figure 5.4: Comparison between the number of points generated by MC and the UT with a varying number of array elements

The total number of points computed by the UT grows exponentially in a N_{UT}^M manner while the MC has a linear growth. However, the UT can be advantageous since MC simulations may require millions of randomly generated points. In Figure 5.4, it is seen that the usage of the 3-point UT can be advantageous for up to 15 antennas if compared to 10^6 generated positions in a MC fashion. The same is true for 10 antenna elements in the

5-point UT case.

5.4 Simulation Results

In Section 5.3, it is shown how the $\overline{\text{SINR}}$ is computed using the UT and the number of points required for both MC and the UT simulations. In this section, the SINR averages are simulated and plotted.

One fundamental question that arises from figures 5.3 and 5.4 in Section 5.3 is: over how many points do we need to compute MC mean? The answer is not straight forward. The number of MC simulations has a direct impact on the precision of the mean, i.e. how close the sample mean is to the ground truth. The ground truth is hard to compute and, in this case, not known. Therefore, the achieved precision is not verifiable. However, it may be considered that the error of the computed mean as a random variable that also has an associated variance and, of course, the smaller the variance, the better the estimate.

To illustrate the statement above, the light gray line in Figure 5.5 shows the change in the sample variance of the SINR mean from Equation (5.22) after 100 samples for a varying number of trials. The scenario is the same using 5 antennas as described further in this

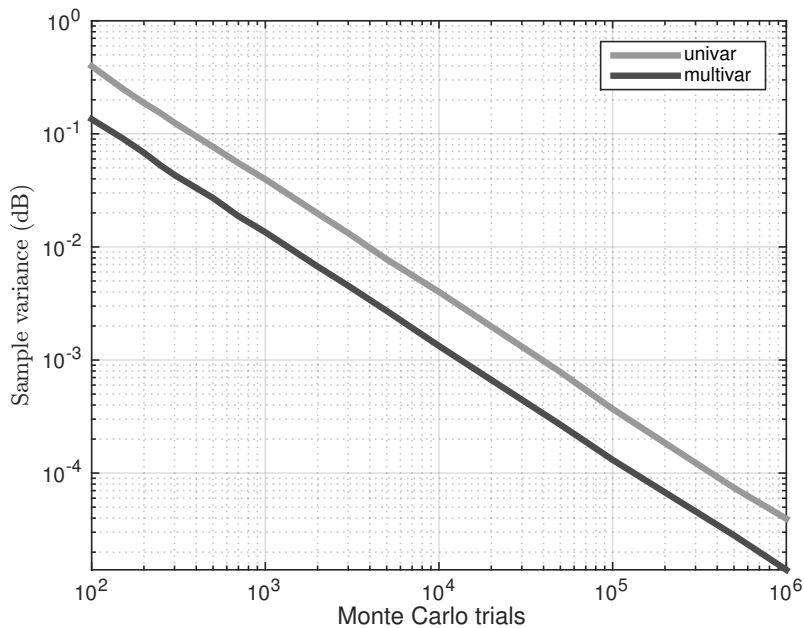


Figure 5.5: Sample variance versus the number of trials in Monte Carlo simulations

section. In Figure 5.5, it is seen that a few tenths of thousands of trials are needed so that the variance is below 10^{-3} dB, i.e. small in comparison to the 10 dB SINR range that is shown in Sections 5.4.1 and 5.4.2. The dark line in Figure 5.5 shows the result for the multivariate case of Equation (5.26). The result shows that the variance is slightly lower for

the multivariate case, as the number of points increases. Therefore, by inspection of Figure 5.5, it can be considered that the variance needed to accomplish a reasonable precision is at least one thousandth of the range in investigation, in this case, 10^{-3} dB.

For the simulations, 5 antenna elements and 2 impinging signals are considered. Also, one SOI is positioned at 90° and two interfering signals are positioned at 65° and 117° , respectively. The DOAs are then varied and SINR mean is computed using the 3 and 5-point UT and 5 and 10^6 MC realizations. The results are shown in Figure 5.6. In order to

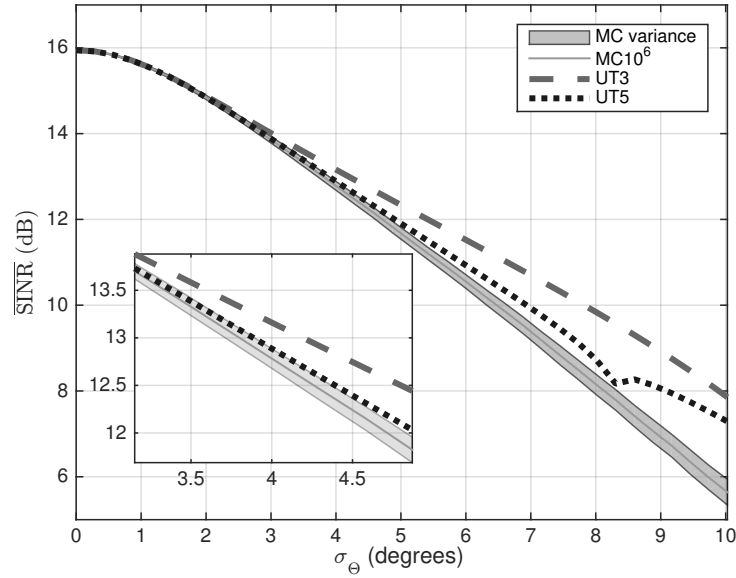


Figure 5.6: $\overline{\text{SINR}}$ vs σ_Θ due to DOA estimation error

guarantee a precise estimate of the mean and after inspecting Figure 5.5, 10^6 MC realizations are used as a reference. From Figure 5.6, it is seen that the $\overline{\text{SINR}}$ estimated using the 3-point UT follows almost exactly the 10^6 MC realizations $\overline{\text{SINR}}$ for up to 3° of standard deviation. The same holds for the 5-point UT for up to $\sigma_\Theta = 4^\circ$. To get a better estimation, more points are needed, however, a standard deviation of 4° is already a large value for most applications. It is also important to remind that the UT computations are done with a small fraction of the computational cost of the MC simulations.

For the simulations, it is assumed that the mean over 10^6 MC realizations, represented by $\overline{\text{SINR}}_{\text{MC}10^6}$, is very close to the ground truth mean. Therefore, the error is defined as

$$\text{Error} = |\overline{\text{SINR}} - \overline{\text{SINR}}_{\text{MC}10^6}|. \quad (5.32)$$

Equation (5.32) is used in Sections 5.4.1 and 5.4.2 for the validation of the proposed UT-based methods.

In Section 5.4.1 the SINR, the error and the computational time of the univariate UT are

evaluated. In Section 5.4.2, the SINR, the error and the computational time are evaluated for the multivariate case.

5.4.1 Performance Evaluation of the Univariate UT

In Figure 5.6, the region between the average SINR and the variance of the estimation is shown by the shaded area. The variance was estimated for 100 trials of 10^5 generated points, where the mean over the 100 trials gives the final 10^6 points average. The shaded region gives a degree of confidence in the estimated MC. At the bottom-left corner, the graphic is zoomed in the region where the UT5 leaves the region of confidence around 4° .

Figure 5.7 shows the relative error between the UTs and the reference MC simulation after 10^6 realizations. From Figure 5.7, it is seen that the 5-point UT still follows the MC 10^6 $\overline{\text{SINR}}$ up to $\sigma_\Theta = 6^\circ$ with a less than 0.5 dB error margin. It is also worth noting that the error does not grow monotonically, however, the growth trend can be clearly seen. In [62], σ_Θ is varied up to 2.63° using 1000 MC trials. In Figure 5.7, it is seen that the UT5 give practically zero error up to this level, therefore, the 5-point UT can be used to compute the $\overline{\text{SINR}}$.

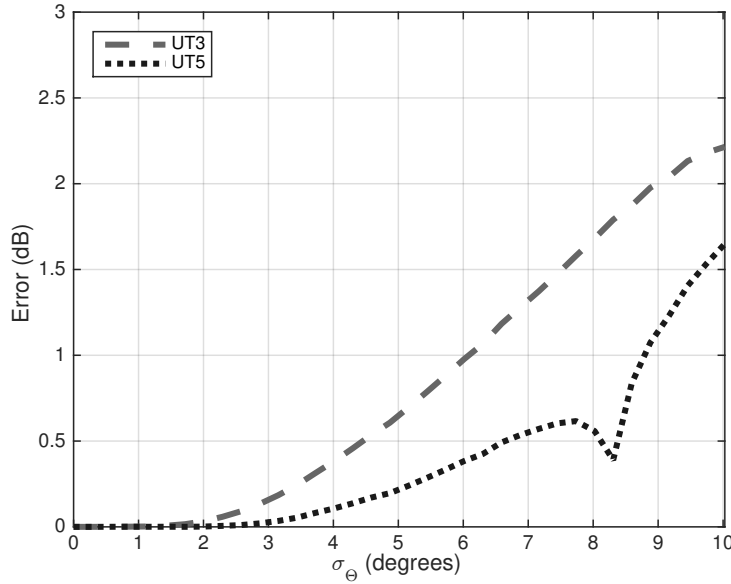


Figure 5.7: Error vs σ_Θ due to DOA estimation errors

The standard deviation is then fixed in two values $\sigma_\Theta = \{1, 6\}^\circ$ and vary the SNR as shown in Figure 5.8. The darker plots show the $\overline{\text{SINR}}$ for $\sigma_\Theta = 1^\circ$ and the brighter ones are for $\sigma_\Theta = 6^\circ$. For a small standard deviation, both UTs follow the MC simulation exactly. For a larger deviation, the UT fits the MC $\overline{\text{SINR}}$ up to an SNR of 2 dB for the 3-point UT and 8 dB for the 5-point UT. The reason is that, for low SNRs, the noise variance on the

denominator of equations (5.22), (5.25), (5.26) and (5.31) is greater than the other terms impact on the SINR than the error due to DOA estimation.

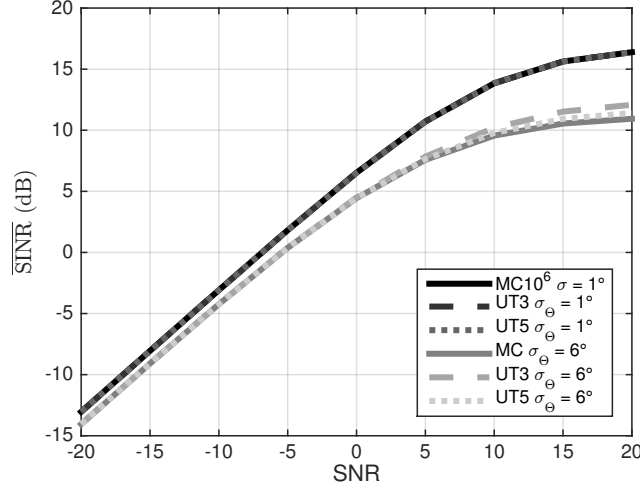


Figure 5.8: $\overline{\text{SINR}}$ vs SNR with fixed $\sigma_{\Theta} = \{2, 10\}^{\circ}$

In the next simulation, the computational time is assessed and the results are shown in Figure 5.9. The 3-point UT and the 5-point UT are compared for a varying number of MC trials. The results show that the 3-point UT and the 5-point UT have a low computational time in comparison to the MC simulations even when only tens of trials are performed. It is clearly seen that these results are tied to the number of points generated as shown in Figure 5.3.

5.4.2 Performance Evaluation of the Multivariate UT

In Figure 5.10, the multivariate case with fixed SINR and a varying $\sigma_{\mathcal{D}}$ is simulated for random element positions. Again, the SINRs averaged using the 3-point UT follows the MC curve very closely for small deviations. For larger deviations, the 5-point UT fits the MC $\overline{\text{SINR}}$. Such large deviations are, however, improbable to occur in an array manufacturing process. In a case where the deviation is large, e.g. cooperative array [2], the MC curve can be followed further by increasing the number of points, although care should be taken due to the rapid increase in complexity in multivariate scenarios (see Figure 5.4). For comparison, a comparable computational complexity with the one in Figure 5.4 is used. As in Figure 5.6, the region between the variances is shaded. Also, the zoomed curves are shown in the top-right corner focusing the point where the UT5 leaves the shaded region around 0.09λ .

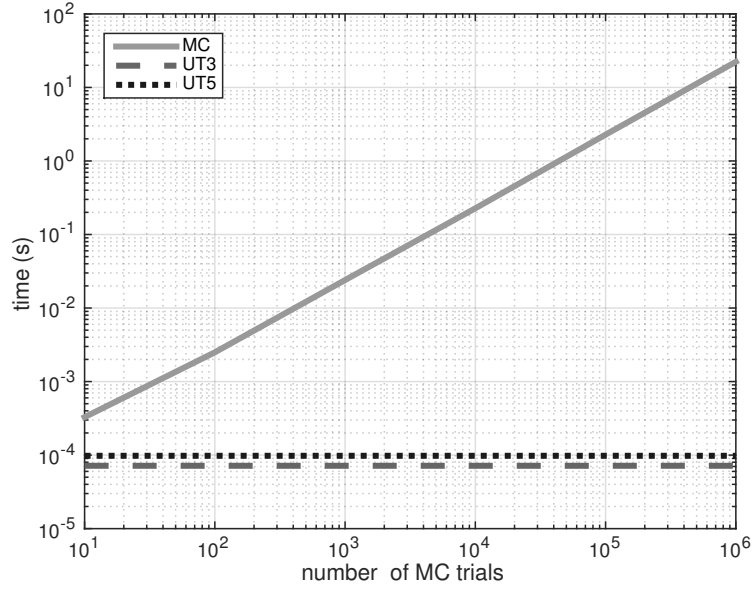


Figure 5.9: Computational time of the univariate UT in the presence of DOA estimation error

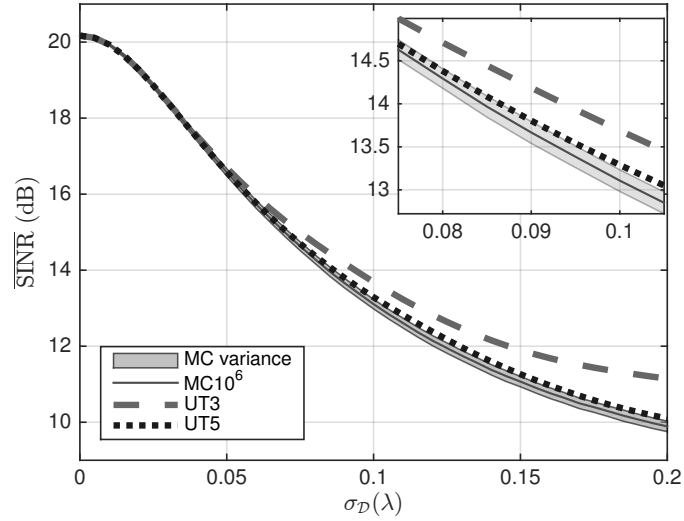


Figure 5.10: $\overline{\text{SINR}}$ vs σ_D due to array element positioning imperfections

Figure 5.11 shows the error for the multivariate case. From Figure 5.11 it is verified that the error is virtually zero for the 3 and 5-point UT with a standard deviation of up to 0.03 and 0.05, respectively. Also, the 3-point UT error grows faster than the 5-point UT as expected. Similarly as in the previous scenario, one may not expect a monotonic growth in the error computation for a increasing σ_D . The error for the 625 realizations SINR average varies largely. Therefore, it is not possible to conclude the error levels.

In Figure 5.12, the standard deviation σ_D is fixed to $\{0.03, 0.12\}$ and the SNR is varied.

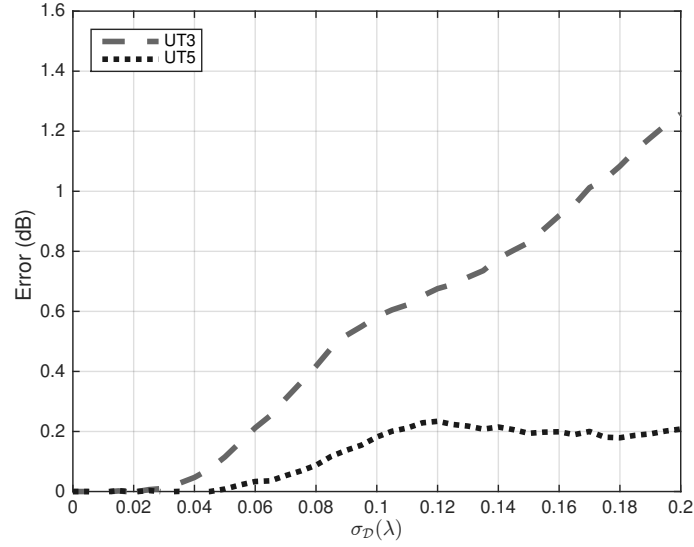


Figure 5.11: Error versus σ_D due to array element positioning imperfections

Similarly as in the univariate case, the UT3 and UT5 $\overline{\text{SINR}}$ s follow the MC SINR average. When the noise is low and the positioning error is high, the UT $\overline{\text{SINR}}$ s deviate from the MC mean. However, this deviation is small and the positioning error is high most of the practical applications.

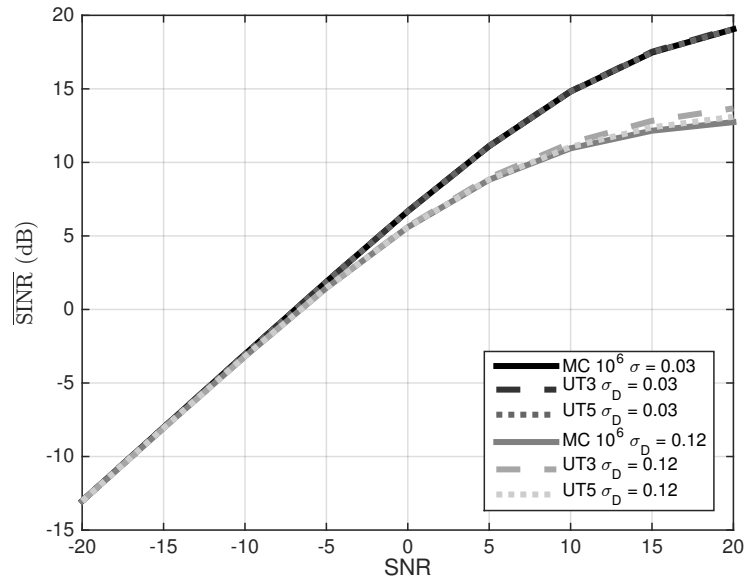


Figure 5.12: $\overline{\text{SINR}}$ vs SNR with fixed $\sigma_D = \{0.03, 0.12\}$

The final simulation assesses the computational time of the multivariate UT and the results are depicted in Figure 5.13. The 3-point UT and the 5-point UT are compared for a varying number of MC trials. The results indicate that the 3-point UT has a low computational time in comparison to the MC simulations when roughly more than 100 MC

trials are needed. The 5-point UT is advantageous from the computational point of view when roughly more than 1000 MC trials are required. This result is in accordance with the number of points that need to be generated for the UT and MC simulations as shown in Figure 5.4.

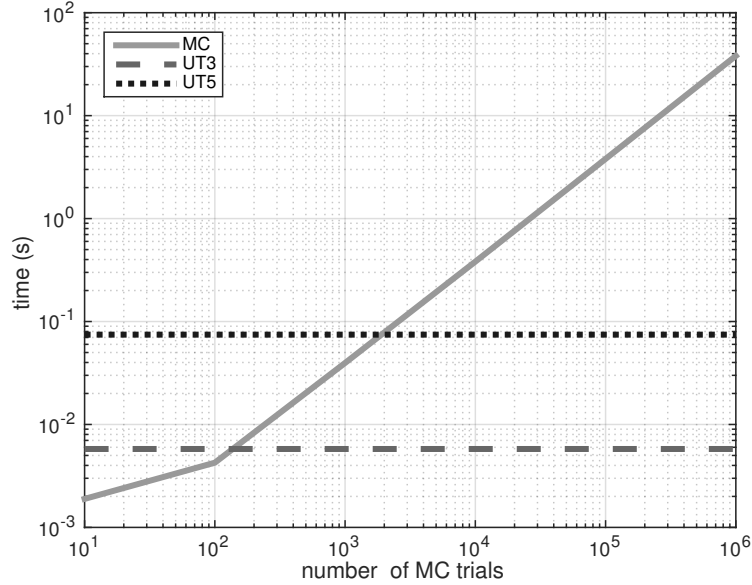


Figure 5.13: Computational time of the multivariate UT in the case of element positioning perturbations

5.5 Summary

This chapter shows a novel method that used the UT for assessing the SINR average of a beamformer. It is shown that the computational burden of the evaluation of a beamformer can be greatly reduced while maintaining the reliability of the simulations. The DOA estimation error and imperfections of the array elements positions are chosen in order to assess the SINR average of a DS beamformer using the proposed method and compared with the traditional MC simulations. The simulations shows that the UT is very reliable even only 5 points are computed for small to medium error.

VI

CONCLUSIONS

6.1 General Conclusions

Array signal processing is already present in the everyday life of modern society. This presence, however is still restricted to a few type of communication devices or to a few, not more than a handful, antennas. Proposed requirements for the next generation mobile networks suggests that these networks will support tens to hundreds of antennas. Also, to allow full coverage, the usage of small satellites to inter-operate with the 5G network is being proposed. Current mobile networks have difficulties to reach mountains and other remote areas. By inserting satellites into these networks such problem should be overcome.

When it comes to array signal processing, beamforming is a key topic. In previous works, usually only simplistic scenarios containing only white noise are considered. However, in real applications this is usually not the case. In this work we first investigated prewhitening techniques to mitigate the introduction of colored noise into the signal model. Moreover, other techniques enhance the interference mitigation were used such as rank reduction and the VIT. These techniques have better effect when the number of antennas is large, i.e. they should work seamlessly in next generation systems.

When applying beamforming to multidimensional antenna arrays, the data is usually vectorized prior to filter estimation. By vectorizing the multidimensionality of the data is not explored. In this work the multidimensionality of the data was explored in GSC beamformers. It was seen that the classical approach is equivalent to using the highest or the R -th dimension unfolding. In the proposed algorithm, all the $R - 1$ unfoldings to achieve better interference mitigation.

It is possible to apply frequency invariant transformations to allow for broadband signal processing. By usage of a bank of FIBs the transformation is achieved and the extracted signal can be processed via PARAFAC decomposition. The decomposition used explores the non-whiteness of the signals and require a novel manner to construct correlation tensors so

that it can be applied for adaptive PARAFAC decomposition tracking algorithms. Results have a better SINR when compared to state of the art independent component analysis.

6.2 Recommendations for Future Research

For future research it is recommended that RLS-type adaptive algorithms are tested. RLS-type adaptive algorithms are known to have decreased convergence time in exchange for higher complexity. However, for current hardware, this complexity increase may not limit its implementation for many cases. This tests can also be done for colored noise scenarios and along with rank reduction and the VIT. Also the proposed block-wise algorithm works for CM algorithms such as the 4-QAM or QPSK signals. It is also of interest the development of similar solutions for more generic QAM or other type of modulations.

For the multidimensional beamformer, a preliminary research on adaptive algorithms was conducted for an LMS adaptation. However a deeper investigation is on its characteristics is recommended and the RLS adaptation should also be considered. Moreover, the multidimensional GSC has shown to have a bad behavior when any of the dimensions of the array is smaller than the number of sources. Alternatives of resolution or mitigation of this effect may also be considered.

For broadband scenarios, a bank of FIBs where used so that narrowband techniques could be applied. The application of the novel correlation tensor construction for a broadband signals in the frequency domain may also be applied, since the DFT may also be considered as a filter bank. In this case, the permutation problem will occur for each frequency and extra complexity will be added to the algorithm, however the decreased number of antenna elements may be worth by decreasing RF hardware complexity.

A

BLOCKING MATRIX COMPUTATION

This appendix reviews popular methods for computing the blocking matrices [81] found in Chapters II and III.

The blocking matrix has its name for blocking the desired signal. In this process it lets interference only signal pass through the system so that the filter can be adapted in order to form nulls in the interference directions. The blocking matrix is defined as the orthogonal compliment of the desired signals steering matrix yielding $\mathbf{B}\mathbf{A}(\boldsymbol{\theta}) = \mathbf{0}$, where \mathbf{B} is the blocking matrix, \mathbf{A} is the steering matrix and $\boldsymbol{\theta}$ is a vector containing the DOAs of the desired signals. For the purpose of illustration we can use a 3 antenna array so that 3 dimensional space is possible to be drawn. In the case where 2 constrains, or 2 desired signals, are present, we end up with the scenario illustrated in Figure A.1.

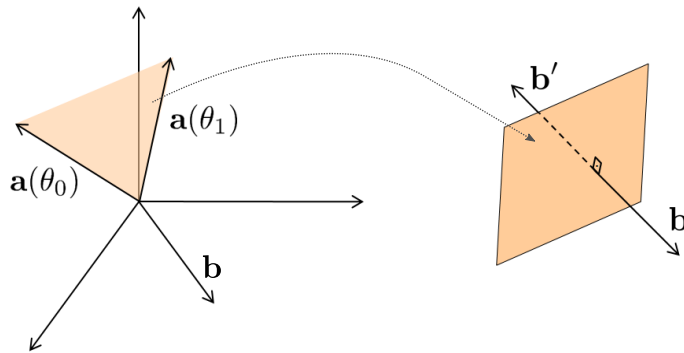


Figure A.1: Illustration of an 1-D blocking “matrix”

In Figure A.1, the desired signal steering vectors span a plane. Therefore, we are left with only one degree of freedom what yields into the blocking matrix being a vector instead. Also note that the solution for \mathbf{b} is not unique since \mathbf{b}' is also orthonormal to the plane. In Sections to three methods on how to find \mathbf{B} will be explained.

Now we get to the practical implementations of the blocking matrix computation. Let us consider one desired signal with steering vector $\mathbf{a}(\theta_0)$. The projection operator of $\mathbf{a}(\theta_0)$ is:

$$\mathbf{P}_{\mathbf{a}(\theta_0)}^\perp = \mathbf{I} - \mathbf{a}(\theta_0)(\mathbf{a}^H(\theta_0)\mathbf{a}(\theta_0))^{-1}\mathbf{a}^H(\theta_0). \quad (\text{A.1})$$

The projection operator (A.1) gives vectors that are orthogonal to $\mathbf{a}(\theta_0)$. In the case where 3 antennas are considered, thus giving us a 3-D space, we can represent is as depicted in Figure A.2.

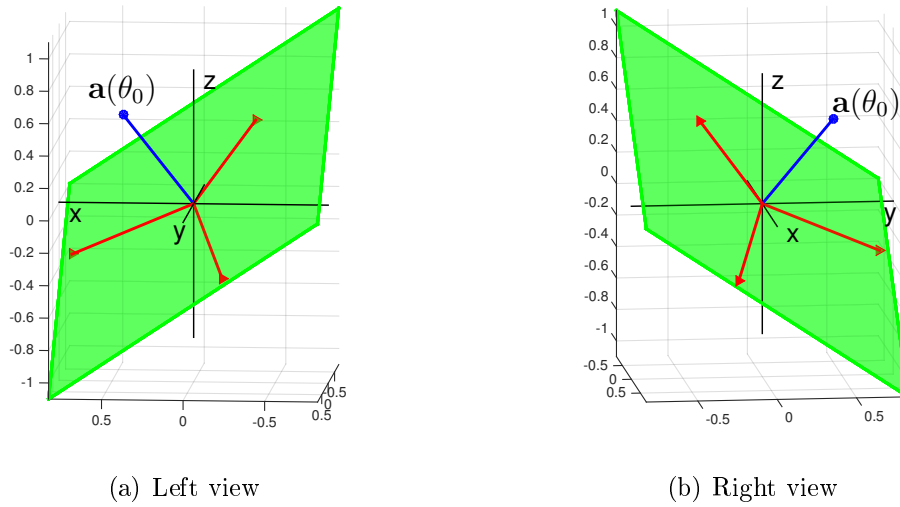


Figure A.2: Projections of a blocking matrix for $M = 3$ and $d = 1$

The desired signal steering vector is represented in blue while the orthogonal plane is shown in green. The red lines are all candidates for the columns of \mathbf{B} . Since there is no unique solution, we might simply choose $d - 1 = 2$ and orthogonalize them using the Gram-Schmidt orthogonalization.

Another way to get the orthogonalized vectors is to take SVD of the projection operator (A.1). Then $d-1$ columns of the right side singular vectors matrix associated with the largest singular values are chosen as illustrated in Figure A.3.

$$\mathbf{P}_{\mathbf{a}(\theta_0)}^\perp = \begin{bmatrix} \mathbf{B} & \end{bmatrix} \begin{bmatrix} \diagdown & \\ & \end{bmatrix} \begin{bmatrix} \end{bmatrix}$$

Figure A.3: Illustration of the right side singular vectors matrix associated with the largest singular values

Another simple method to find \mathbf{B} is to directly take the SVD of $\mathbf{a}(\theta_0)$ and then take the

d-1 columns of the right side singular vectors matrix associated with the smallest singular values. This process is illustrated in Figure A.4.

$$\mathbf{a}(\theta_0) = \begin{bmatrix} \text{white} & \text{B} \end{bmatrix} \begin{bmatrix} \diagdown \end{bmatrix} \begin{bmatrix} \text{white} \end{bmatrix}$$

Figure A.4: Illustration of the right side singular vectors matrix associated with the smallest singular values

B

DERIVATION OF THE BLOCK-WISE STOCHASTIC GRADIENT GSC

This appendix makes the derivation of the block-wise stochastic gradient GSC from Chapter II. We start by expanding the CM cost function (2.14) for the GSC:

$$J_{\text{CM}}(\mathbf{T}, \bar{\mathbf{w}}) = \mathbb{E}\{|\mathbf{a}(\theta_0) - \mathbf{B}^H \mathbf{T} \bar{\mathbf{w}}|^2\} \quad (\text{B.2})$$

$$= \mathbb{E}\{(\mathbf{a}^H(\theta_0) \tilde{\mathbf{x}} - \bar{\mathbf{w}}^H \mathbf{T}^H \mathbf{B} \tilde{\mathbf{x}}) \cdot (\mathbf{a}^H(\theta_0) \tilde{\mathbf{x}} - \bar{\mathbf{w}}^H \mathbf{T}^H \mathbf{B} \tilde{\mathbf{x}})^H\} \quad (\text{B.3})$$

$$\begin{aligned} &= \mathbb{E}\{\mathbf{a}^H(\theta_0) \tilde{\mathbf{x}} \tilde{\mathbf{x}}^H \mathbf{a}(\theta_0) - \bar{\mathbf{w}}^H \mathbf{T}^H \mathbf{B} \tilde{\mathbf{x}} \tilde{\mathbf{x}}^H \mathbf{a}(\theta_0) + \mathbf{a}^H(\theta_0) \tilde{\mathbf{x}} - \\ &\quad \bar{\mathbf{w}}^H \mathbf{T}^H \mathbf{B} \tilde{\mathbf{x}} \tilde{\mathbf{x}}^H \mathbf{a}(\theta_0) + \bar{\mathbf{w}}^H \mathbf{T}^H \mathbf{B} \tilde{\mathbf{x}} \tilde{\mathbf{x}}^H \mathbf{B}^H \mathbf{T} \bar{\mathbf{w}} + \mathbf{a}^H(\theta_0) \tilde{\mathbf{x}} - \\ &\quad \bar{\mathbf{w}}^H \mathbf{T}^H \mathbf{B} \tilde{\mathbf{x}} - \bar{\mathbf{w}}^H \mathbf{T}^H \mathbf{B} \tilde{\mathbf{x}} + 1\} \end{aligned} \quad (\text{B.4})$$

$$\begin{aligned} &= \mathbf{a}^H(\theta_0) \mathbf{R}_{\tilde{\mathbf{x}}\tilde{\mathbf{x}}} \mathbf{a}(\theta_0) - 2\bar{\mathbf{w}}^H \mathbf{T}^H \mathbf{B} \mathbf{R}_{\tilde{\mathbf{x}}\tilde{\mathbf{x}}} \mathbf{a}(\theta_0) + \\ &\quad 2\mathbf{a}^H(\theta_0) \mathbb{E}\{\tilde{\mathbf{x}}\} + \bar{\mathbf{w}}^H \mathbf{T}^H \mathbf{B} \mathbf{R}_{\tilde{\mathbf{x}}\tilde{\mathbf{x}}} \mathbf{B}^H \mathbf{T} \bar{\mathbf{w}} - 2\bar{\mathbf{w}}^H \mathbf{T}^H \mathbf{B} \mathbb{E}\{\tilde{\mathbf{x}}\} + 1 \end{aligned} \quad (\text{B.5})$$

Now we take the gradient with respect to $\bar{\mathbf{w}}$:

$$\begin{aligned} \nabla_{\bar{\mathbf{w}}} J_{\text{CM}} &= -2\mathbf{T}^H \mathbf{B} \mathbf{R}_{\tilde{\mathbf{x}}\tilde{\mathbf{x}}} \mathbf{a}(\theta_0) + 2\mathbf{T}^H \mathbf{B} \mathbf{R}_{\tilde{\mathbf{x}}\tilde{\mathbf{x}}} \mathbf{B}^H \mathbf{T} \bar{\mathbf{w}} - 2\mathbf{T}^H \mathbf{B} \mathbb{E}\{\tilde{\mathbf{x}}\} \\ &= -2\mathbf{T}^H \mathbf{B} (\mathbf{R}_{\tilde{\mathbf{x}}\tilde{\mathbf{x}}} \mathbf{B}^H \mathbf{T} \bar{\mathbf{w}} - \mathbf{R}_{\tilde{\mathbf{x}}\tilde{\mathbf{x}}} \mathbf{a}(\theta_0) - \mathbb{E}\{\tilde{\mathbf{x}}\}) \end{aligned} \quad (\text{B.6})$$

And with respect to \mathbf{T} :

$$\nabla_{\mathbf{T}} J_{\text{CM}} = -2\mathbf{B} \mathbf{R}_{\tilde{\mathbf{x}}\tilde{\mathbf{x}}} \mathbf{a}(\theta_0) \bar{\mathbf{w}}^H + \mathbf{B} \mathbf{R}_{\tilde{\mathbf{x}}\tilde{\mathbf{x}}} \mathbf{B}^H \mathbf{T} \bar{\mathbf{w}} \bar{\mathbf{w}}^H - 2\mathbf{B} \mathbb{E}\{\tilde{\mathbf{x}}\} \bar{\mathbf{w}}^H \quad (\text{B.7})$$

Since we have a whole block stored in memory, we try to use all the information available to build the stochastic gradients. We compute the correlation and the expected values of $\tilde{\mathbf{x}}$

as:

$$\hat{\mathbf{R}}_{\tilde{\mathbf{x}}\tilde{\mathbf{x}}} = \frac{1}{L} \tilde{\mathbf{X}} \tilde{\mathbf{X}}^H, \quad (\text{B.8})$$

$$\hat{\mathbf{E}}\{\tilde{\mathbf{x}}\} = \frac{1}{L} \sum_{n=mL}^{(m+1)L-1} \tilde{\mathbf{x}}(n) = \frac{1}{L} \tilde{\mathbf{X}} \cdot \mathbf{1}_L, \quad (\text{B.9})$$

where $\mathbf{1}_L$ is a column vector of ones of size L . The matrix $\tilde{\mathbf{X}}$ is the matrix containing N time samples of the $\tilde{\mathbf{x}}(n)$ vector. Extending the definition $\tilde{\mathbf{x}}(n) = y^*(n)\mathbf{x}(n)$ for the matrix notation we have that:

$$\tilde{\mathbf{X}} = \mathbf{X} \cdot \text{diag}(\mathbf{e}_m), \quad (\text{B.10})$$

where $\mathbf{e}_m = [e(mL), e(mL+1), \dots, e((m+1)L-1)]^H$. Then we insert the approximations into (A.5) for $\bar{\mathbf{w}}$:

$$\begin{aligned} \hat{\nabla}_{\bar{\mathbf{w}}} J_{\text{CM}} &= 2 \frac{\mathbf{T}^H \mathbf{B}}{L} (\tilde{\mathbf{X}} \tilde{\mathbf{X}}^H \mathbf{B}^H \mathbf{T} \bar{\mathbf{w}} - \tilde{\mathbf{X}} \tilde{\mathbf{X}}^H \mathbf{a}(\theta_0) - \sum_{n=mL}^{(m+1)L-1} \tilde{\mathbf{x}}(n)) \\ &= 2 \frac{\mathbf{T}^H \mathbf{B}}{L} (\tilde{\mathbf{X}} \mathbf{y}_m^H - \tilde{\mathbf{X}} \mathbf{d}_m^H - \sum_{n=mL}^{(m+1)L-1} \tilde{\mathbf{x}}(n)) \\ &= -\frac{2}{L} \mathbf{T}^H \mathbf{B} (\tilde{\mathbf{X}} \mathbf{e}_m + \tilde{\mathbf{X}} \cdot \mathbf{1}_L) \\ &= -\frac{2}{L} \mathbf{T}^H \mathbf{B} \tilde{\mathbf{X}} (\mathbf{e}_m + \mathbf{1}_L), \end{aligned} \quad (\text{B.11})$$

where the vectors \mathbf{y}_m and \mathbf{d}_m are vectors over the time dimension, i.e. column vectors, with the samples from the m -th time block. We also insert the approximations into (A.6) for \mathbf{T} :

$$\begin{aligned} \hat{\nabla}_{\mathbf{T}} J_{\text{CM}} &= -2 \mathbf{B} \frac{\tilde{\mathbf{X}} \tilde{\mathbf{X}}^H}{L} \mathbf{a}(\theta_0) \bar{\mathbf{w}}^H + 2 \mathbf{B} \frac{\tilde{\mathbf{X}} \tilde{\mathbf{X}}^H}{L} \mathbf{B}^H \mathbf{T} \bar{\mathbf{w}} \bar{\mathbf{w}}^H - 2 \mathbf{B} \frac{1}{L} [\tilde{\mathbf{X}} \cdot \mathbf{1}_L] \bar{\mathbf{w}}^H \\ &= -\frac{2}{L} \mathbf{B} \tilde{\mathbf{X}} (\mathbf{e}_m + \mathbf{1}_L) \bar{\mathbf{w}}^H \end{aligned} \quad (\text{B.12})$$

This leads to the block-wise "joint" update rule

$$\mathbf{T}(n+1) = \mathbf{T}(n) + \mu_T \tilde{\mathbf{X}}_B (\mathbf{e}_m + \mathbf{1}_L) \bar{\mathbf{w}}^H \quad (\text{B.13})$$

$$\bar{\mathbf{w}}(n+1) = \bar{\mathbf{w}}(n) + \mu_w \mathbf{T}^H \tilde{\mathbf{X}}_B (\mathbf{e}_m + \mathbf{1}_L), \quad (\text{B.14})$$

where μ_w and μ_T are represented in terms of $1/L$. If we define $\mathbf{e}_m = \mathbf{e}_m + \mathbf{1}_L$ we can further simplify the update equations

$$\mathbf{T}(n+1) = \mathbf{T}(n) + \mu_T \tilde{\mathbf{X}}_B \mathbf{e}_m \bar{\mathbf{w}}^H \quad (\text{B.15})$$

$$\bar{\mathbf{w}}(n+1) = \bar{\mathbf{w}}(n) + \mu_w \mathbf{T}^H \tilde{\mathbf{X}}_B \mathbf{e}_m, \quad (\text{B.16})$$

List of Tables

II.1	Computational costs in flops	36
II.2	Notation of the legends used in the figures	38
IV.1	Proposed FIB-RLST-Ben and FIB-RLST-Frost	75
IV.2	Parameters for the first scenario	75

List of Figures

1.1	Senusoidal waves were generated at 5 Hz and virtually steered, for the sake of visualization of the beamforming. The constructive patterns are clearly visible and appear in dark red.	17
1.2	Illustration of a matrix of size $M \times N$ and 3-dimensional tensor of size $M \times N \times P$	18
2.1	Batch GSC block diagram	26
2.2	LMS-GSC block diagram	27
2.3	RR-LMS-GSC block diagram	28
2.4	High accuracy stochastic prewhitening block diagram	30
2.5	VIT-HASP block diagram	32
2.6	Deterministic prewhitening diagram.	33
2.7	RMSE of the LMS-GSC type algorithms versus samples for $N = 50$ and an SNR of 10 dB	37
2.8	RMSE of the RR-LMS-GSC type algorithms versus samples for $N = 50$ and an SNR of 10 dB	39
2.9	RMSE for LMS-GSC type algorithms with a varying N and an SNR of 10 dB	40
2.10	Final RMSE for algorithms with a varying N , an SNR of 10 dB and $\rho = 0.9$.	41
2.11	Final RMSE after $N = 600$ snapshots with a varying ρ and an SNR of 10 dB	41
3.1	Elevation ϕ_i and azimuth θ_i of the direction of arrival of the i -th source . . .	44
3.2	Using a URA to form a 3-D data tensor	45
3.3	Classical GSC for R dimensions.	46
3.4	Block diagram of the R -D GSC algorithm	47
3.5	Magnitude of the spatial response of the GSC filter	49
3.6	Effective SINR after applying the Classical and proposed 2-D GSC for a varying number of temporal snapshots N	51
3.7	Effective SINR after applying the Classical and proposed 2-D GSC for a varying SNR	51
3.8	Effective SINR after applying the Classical and proposed 2-D GSC for a varying array size. Both dimensions lengths are kept equal	52

3.9	Computational time versus N	52
4.1	Scenario cointaing a URA and a ULA. Each signal has a single path to each antenna.	56
4.2	Tap-delay line. Each antenna m receives a broadband signal filtered by an FIR filter containing L taps.	57
4.3	Spatial and frequency response of \mathbf{w} steered to 0°	59
4.4	Region of existence of the spatio-temporal response	61
4.5	Spatial and frequency response of a FIB	62
4.6	Taps of the filter $\mathbf{w}[m, n]$	63
4.7	Responses of FIBs inside the frequency invariant region; (a) gain at $\sin(\theta) = -0.4$ is almost -35 dB and 0 dB at $\sin(\theta) = 0$. (b) the gain at $\sin(\theta) = 0$ is roughly -30 dB and 0 dB at $\sin(\theta) = -0.4$	63
4.8	Diagram of a filtering scheme composed of a bank of FIBs and narrowband adaptive algorithm	64
4.9	Response of a bank of 5 FIBs at a frequency inside the invariant range . . .	65
4.10	Illustration of a PARAFAC decomposition of a 3-dimensional (or three-way) array	66
4.11	Concatenation of a new slice in the second dimension for a tensor with 3 dimensions	69
4.12	Process of selecting, removing and appeding a new slice to the PARAFAC RLST input correlation tensor. After $\mathbf{R}(t + J - 1)$ is updated, the process is restarted and the tensor is updated of with the new correlation matrix for the first time-delay τ_1	71
4.13	SINR for QPSK signals	76
4.14	Spatial response over the invariant frequency band	77
4.15	SINR for a varying DOA	77
4.16	Beam response after $N = 5000$ of the proposed FIB-RLST-Ben filter	78
4.17	Beam response after $N = 10000$ of the proposed FIB-RLST-Ben filter	78
4.18	SINR comparison between ICA and the PARAFAC-based algorithms. The result is filtered by a moving average of length 10 for clarity.	79
4.19	Beampattern of the resulting FIB-RLST-Ben beamformer	79
5.1	Illustration of a plane wavefront impinging on an antenna array containing DOA and element positioning errors	83
5.2	Comparison between a continuous PDF and a 3-point UT and 5-point UT PDF approximation	86
5.3	Number of generated random samples	89

5.4	Comparison between the number of points generated by MC and the UT with a varying number of array elements	91
5.5	Sample variance versus the number of trials in Monte Carlo simulations . . .	92
5.6	$\overline{\text{SINR}}$ vs σ_{Θ} due to DOA estimation error	93
5.7	Error vs σ_{Θ} due to DOA estimation errors	94
5.8	$\overline{\text{SINR}}$ vs SNR with fixed $\sigma_{\Theta} = \{2, 10\}^{\circ}$	95
5.9	Computational time of the univariate UT in the presence of DOA estimation error	96
5.10	$\overline{\text{SINR}}$ vs $\sigma_{\mathcal{D}}$ due to array element positioning imperfections	96
5.11	Error versus $\sigma_{\mathcal{D}}$ due to array element positioning imperfections	97
5.12	$\overline{\text{SINR}}$ vs SNR with fixed $\sigma_{\mathcal{D}} = \{0.03, 0.12\}$	97
5.13	Computational time of the multivariate UT in the case of element positioning perturbations	98
A.1	Illustration of an 1-D blocking “matrix”	101
A.2	Projections of a blocking matrix for $M = 3$ and $d = 1$	102
A.3	Illustration of the right side singular vectors matrix associated with the largest singular values	102
A.4	Illustration of the right side singular vectors matrix associated with the smallest singular values	103

SYMBOLS

a, b, c	Scalars
$\mathbf{a}, \mathbf{b}, \mathbf{c}$	Vectors
$\mathbf{A}, \mathbf{B}, \mathbf{C}$	Matrices
$\mathcal{A}, \mathcal{B}, \mathcal{C}$	Tensors
\cdot^*	Conjugate of a Scalar, Vector, Matrix or Tensor
\times_r	r -th mode tensor Product
$[\mathcal{X}]_{(r)}$	r -th mode tensor unfolding
$[\mathcal{X}]_{(r)}^{\odot}$	r -th mode forward cyclical tensor unfolding
$\nabla \cdot$	Gradient
$ \cdot $	Absolute Value
$\ \cdot\ _{\text{F}}^2$	Frobenius Norm
$\ \cdot\ _{\text{H}}^2$	Higher Order Frobenius Norm of a Tensor
$\lceil \cdot \rceil$	Round up
$\lfloor \cdot \rfloor$	Round down
\diamond	Khatri-Rao Product
\otimes	Kronecker Product
$\text{vec}\{\cdot\}$	Vectorize a Matrix
$\text{unvec}\{\cdot\}$	Unvectorize a Vector
$\text{diag}(\cdot)$	Extract Diagonal to a Vector
$\text{E}\{\cdot\}$	Expected Value

\cdot^T	Transpose of a Matrix
\cdot^+	Pseudo Inverse of a Matrix
\cdot^{-1}	Inverse of a Matrix
\cdot^H	Conjugate Transpose of a Matrix
$\langle \cdot \rangle_t$	Time Average

ACRONYMS

4G	Fourth Generation
5G	Fifth Generation
ALS	Alternate Least Squares
CM	Constant Modulus
DFT	Discrete Fourier Transform
DOA	Direction of Arrival
DS	Delay and Sum
FIB	Frequency Invariant Beamformer
GSC	Generalized Sidelobe Canceller
ICA	Independent Component Analysis
IDFT	Inverse Discrete Fourier Transform
IDFT2	Bidimensional Inverse Discrete Fourier Transform
IoT	Internet of things
LCMV	Linearly Constrained Minimum Variance
LMS	Least Mean Squares
LTE	Long Term Evolution
MC	Monte Carlo
MIMO	Multiple Input Multiple Output
MVDR	Minimum Variance Distortionless Response
PARAFAC	Parallel Factor Analysis
PDF	Probability Density Function
QPSK	Quadrature Phase Shift Keying
<i>R</i>-D	<i>R</i> -Dimensional
RLS	Recursive Least Squares
RLST	Recursive Least Squares Tracking
RR	Reduced Rank
SINR	Signal to Interference and Noise Ratio

SNR	Signal to Noise Ratio
SOI	Signal of Interest
ULA	Uniform Linear Array
URA	Uniform Rectangular Array
UT	Unscented Transformation
V2I	Vehicle to Infrastructure
V2V	Vehicle to Vehicle
VANET	Vehicular adhoc Networks
VIT	Vandermonde Invariance Transformation
VR	Virtual Reality

BIBLIOGRAPHY

Publications as First or Co-Author

- [1] D. V. de Lima, J. P. C. L. da Costa, F. Antreich, R. K. Miranda, and G. Del Galdo. High resolution time-delay estimation via direction of arrival estimation and khatri-rao factorization for multipath mitigation. *Workshop on Smart Antennas (WSA), Berlin, 2017.*
- [2] E. P. Freitas, R. K. Miranda, M. A. M. Marinho, J. P. C. L. da Costa, and C. E. Pereira. A practical implementation of a cooperative antenna array for wireless sensor networks. *in Proc. 15th International Conference on Next Generation Wired/Wireless Networking NEW2AN, St. Petersburg 2015.*
- [3] P. F. C. Lima, R. K. Miranda, J. P. C. L. da Costa, R. Zelenovsky, Y. Yuan, and G. Del Galdo. Low complexity blind separation technique to solve the permutation ambiguity of convolutive speech mixtures. *in Proc. 10th International Conferences on Signal Processing and Communication Systems, Gold Coast, 2016.*
- [4] R. K. Miranda, J. P. C. L. da Costa, and F. Antreich. High accuracy and low complexity adaptive generalized sidelobe cancelers for colored noise scenarios. *Elsevier Digital Signal Processing, vol. 34, pp. 48-55, Nov. 2014.*
- [5] R. K. Miranda, J. P. C. L. da Costa, T. Eishima, Y. Nakamura, G. Del Galdo, and F. Roemer. Procedures for integrating, testing and operating advanced microsatellites. *in Proc. 4th IFAC Symposium on Telematics Applications, 2016, Porto Alegre. In Proc. 4th IFAC Symposium on Telematics Applications, 2016.*
- [6] R. K. Miranda, J. P. C. L. da Costa, M. A. M. Marinho, E. P. de Freitas, and R. T. de Sousa Jr. Evaluation of space-time-frequency (stf)-coded mimo-ofdm systems in

realistic channel models. *28th IEEE International Conference on Advanced Information Networking and Applications (AINA-2014) Victoria, Canada, May 2014.*

- [7] R. K. Miranda, J. P. C. L. da Costa, F. Roemer, F. Raschke, T. Eishima, Y. Nakamura, and G. Del Galdo. Implementation of improved software defined radio modulation scheme and command and telemetry software interface for small satellites in 5g systems. *In Proc. of the 19th International Conference on OFDM and Frequency Domain Techniques, 2016.*
- [8] R. K. Miranda, J. P. C. L. da Costa, F. Roemer, L. R. A. X. Menezes, G. Del Galdo, and A. L. F. de Almeida. Low complexity performance assessment of a sensor array via unscented transformation. *Elsevier Digital Signal Processing, vol. 63, pp. 190-198, April 2017.*
- [9] R. K. Miranda, S. K. Sahoo, R. Zelenovsky, and J. P. C. L. da Costa. Improved frequency domain blind source separation for audio signals via direction of arrival knowledge. *in Proc. SDPS 2013, 2013, Campinas. Smart Health Cities, 2013.*
- [10] P. M. G. I. Reis, J. P. C. L. da Costa, R. K. Miranda, and G. Del Galdo. Audio authentication using the kurtosis of esprit based enf estimates. *in Proc. 10th International Conference on Signal Processing and Communication Systems, 2016, Gold Coast.*
- [11] P. M. G. I. Reis, J. P. C. L. da Costa, R. K. Miranda, and G. Del Galdo. Esprit-hilbert based audio tampering detection with svm classifier for forensic analysis via electrical network frequency. *IEEE Transactions on Information Forensics and Security, vol. 12, issue 4, pp. 853 - 864, April 2017.*

References by Other Authors

- [12] Ericsson AB. Ericsson mobility report. *Ericsson AB, Revision A, Stockholm, Sweden, 2016.*
- [13] S. Affes, S. Gazor, and Y. Grenier. An algorithm for multisource beamforming and multi-target tracking. *IEEE Transactions on Signal Processing, vol 44, pp. 1512-1522, 1996.*
- [14] S. M. Alamouti. A simple transmit diversity technique for wireless communications. *IEEE Journal on Selected Areas in Communications, vol. 16, Issue 8, pp. 1451-1458, Oct. 1998.*

-
- [15] I. Arasaratnam and S. Haykin. Cubature kalman filters. *IEEE Transactions on Automatic Control*, vol. 54, no. 6, June 2009.
 - [16] A. Ben-Israel and D. Cohen. On iterative computation of generalized inverses and associated projections. *SIAM Journal on Numerical Analysis*, vol. 3, issue 3, pp. 410 - 419, 1966.
 - [17] J. J. Blanz, A. Papathanassiou, M. Haardt, I. Furió, and P. W. Baier. Smart antennas for combined doa and joint channel estimation in time-slotted cdma mobile radio systems with joint detection. *IEEE Trans. on Vehicular Technology*, vol. 49, no. 2, pp. 293-306, Mar. 2000.
 - [18] Jr. C. D. Meyer. Generalized inversion of modified matrices. *SIAM Journal on Applied Mathematics*, vol. 24, no. 3, pp. 315 - 323, May 1973.
 - [19] J. Capon. High-resolution frequency-wavenumber spectrum analysis. *In Proc. IEEE*, vol. 57, no. 8, pp- 1408 -1418, Aug. 1969.
 - [20] B.D. Carlson. Covariance matrix estimation errors and diagonal loading in adaptive arrays. *IEEE Trans. on Aerospace and Electronic Systems*, vol.24(4), pp.397-401, July 1988.
 - [21] J. D. Carroll and J. J. Chang. Analysis of individual differences in multidimensional scaling via an n-way generalization of ‘eckart-young’ decomposition. *Psychometrika*, vol. 35, pp. 283-319, 1970.
 - [22] Y. H. Chen and Y. S. Lin. Doa estimation by fourth-order cumulants in unknown noise environments. *in Proc. of IEEE ICASSP-93*, pp. 296-299, vol. 4, Minneapolis, USA, Apr. 1993.
 - [23] S. J. Chern, C. H. Sun, and C. C. Chang. Blind adaptive ds-cdma receivers with sliding window constant modulus gsc-rls algorithm. *International Symposium on Intelligent Signal Processing and Communications, ISPACS '06.* , pp. 979-982, Dec. 2006.
 - [24] A. Cichocki and S. Amari. Adaptive blind signal and image processing. *John Wiley & Sons*, 2003.
 - [25] J. P. C. L. da Costa. Parameter estimation techniques for multi-dimensional array signal processing. *Shaker Publisher, Aachen, Germany*, Mar. 2010.
 - [26] J. P. C. L. da Costa, K. Liu, H. C. So, F. Roemer, M. Haardt, and S. Schwarz. Multi-dimensional prewhitening for enhanced signal reconstruction and parameter estimation

- in colored noise with kronecker correlation structure. *Signal Processing, Elsevier publisher, vol. 93, i. 11, pp. 3209-3226, Nov 2013.*
- [27] J. P. C. L. da Costa, F. Roemer, and R. T. de Souza Jr. Enhanced model order estimation in colored noise scenarios via noise prewhitening. *XXIX Simpósio Brasileiro de Telecomunicações (SBrT'11), Curitiba, Brazil.*
- [28] J. P. C. L. da Costa, F. Roemer, and M. Haardt. Deterministic prewhitening to improve subspace parameter estimation techniques in severely colored noise environments. *in Proc. of 54th International Scientific Colloquium (IWK'09), Ilmenau, Germany, Sept. 2009.*
- [29] J. P. C. L. da Costa, F. Roemer, M. Haardt, and R. T. de Sousa Jr. Multi-dimensional model order selection. *EURASIP Journal on Advances in Signal Processing 2011, vol. 26, 20 July 2011, Springer publisher.*
- [30] J. P. C. L. da Costa, A. Thakre, F. Roemer, and M. Haardt. Comparison of model order selection techniques for high-resolution parameter estimation algorithms. *in Proc. 54th International Scientific Colloquium (IWK), Ilmenau, Germany, Sept. 2009.*
- [31] L. R. A. X. de Menezes, A. Ajayi, C. Christopoulos, P. Sewell, and G. A. Borges. Extracting statistical moments of output quantities from a small number of time-domain simulations. *Workshop on Computational Electromagnetics in Time-Domain, CEM-TD 2007, 2007.*
- [32] L. R. A. X. de Menezes, E. A. Jr., M. N. de Sousa, and G. A. Borges. Estimation of the probability density function in electromagnetic propagation problems with the unscented transform and tlm. *Proceedings of The 24th Annual Review of Progress in Applied Computational Electromagnetics, 2008.*
- [33] ETSI. User Equipment (UE) Radio Access Capabilities. *3GPP TS 36.306 version 12.6.0 Release 12.*
- [34] B. P. Flanagan and K. L. Bell. Improved array self calibration with large sensor position errors for closely spaced sources. *Proceedings of the 2000 IEEE Sensor Array and Multichannel Signal Processing Workshop, pp. 484-488, 2000.*
- [35] O. L. Frost. An algorithm for linearly constrained adaptive array processing. *Proceedings of the IEEE, vol. 60, issue 8, 1972.*
- [36] W. R. Gabriel. Using spectral estimation techniques in adaptive army systems. *in Proc. of the Phase Arrays 1985 Symposium, pp. 109: also in-house report. RADC-TR-85-171. Rome Air Development Center, Griffis Air Force Base, Rome, N.Y., Aug. 1985.*

-
- [37] L. C. Godaram. Application of antenna arrays to mobile communications, part ii: Beamforming and direction-of-arrival considerations. *Proceedings of the IEEE*, vol. 85, No. 8, pp. 1195-1245, Aug. 1997.
 - [38] G. H. Golub and Charles F. Van Loan. Matrix computations. *JHU Press*, 1996.
 - [39] X.-F. Gong and Q.-H. Lin. Spatially constrained parallel factor analysis for semi-blind beamforming. in *Proc. of 2011 Seventh International Conference on Natural Computation (ICNC)*, vol. 1, pp. 416-420, July 2011.
 - [40] L. J. Griffiths and C.W. Jim. An alternative approach to linearly constrained adaptive beamforming. *IEEE Trans. on Antennas and Propagation*, vol. 30, no. 1, pp. 27-34, Jan. 1982.
 - [41] J. A. Gubner. Probability and random processes for electrical and computer engineers. 1st ed., *Cambridge University Press*, 2006.
 - [42] M. Viberg H. Krim. Two decades of array signal processing research: the parametric approach. *IEEE Signal Processing Magazine*, vol. 13, Issue 4, pp. 67-94, 1996.
 - [43] M. Haardt, F. Roemer, and G. Del Galdo. Higher-order svd based subspace estimation to improve the parameter estimation accuracy in multi-dimensional harmonic retrieval problems. *IEEE Trans. Signal Processing*, vol. 56, no. 7, pp. 3198-3213, July 2008.
 - [44] M. Haardt, R. S. Thomä, and A. Richter. Multidimensional high-resolution parameter estimation with applications to channel sounding. *High-Resolution and Robust Signal Processing*, Eds. 2004, pp. 255-338, New York, NY, Chapter 5.
 - [45] P. C. Hansen and S. H. Jensen. Prewhitening for rank-deficient noise in subspace methods for noise reduction. *IEEE Trans. Signal Processing*, vol. 53, pp. 3718-3726, Oct. 2005.
 - [46] R. A. Harshman. Foundations of the parafac procedure: Models and conditions for an “explanatory” multi-modal factor analysis. *UCLA working papers in phonetics*, vol. 16, pp. 1-84, 1970.
 - [47] S. Haykin. Adaptive filter theory. 3rd edition, *Prentice-Hall*, 1996.
 - [48] W. Herbordt and W. Kellermann. Adaptive beamforming for audio signal acquisition. *Springer*, pp. 155-194, 2003.
 - [49] G. W. Hill. Algorithm 518: Incomplete bessel function i0. the von mises distribution. *ACM Transactions on Mathematical Software (TOMS)*, vol. 3, no. 3, Sept. 1977.

- [50] A. Hyvärinen, J. Krhunen, and E. Oja. Independent component analysis. *Wiley, 2001*.
- [51] M. T. Ivrlac, T. P. Kupjuhn, and J. A. Nossek. Vandermonde invariance transformation. *in Proc. ICASSP 2001, vol. 5, pp. 2929-2932, Salt Lake City, USA, May 2001*.
- [52] S. J. Julier and J. K. Uhlmann. A general method for approximating nonlinear transformations of probability distributions. *Technical report, RRG, Dept. of Eng. Science, Univ. of Oxford, 1996*.
- [53] S. J. Julier and J. K. Uhlmann. Unscented filtering and nonlinear estimation. *Proceedings of the IEEE, vol. 92, no. 3, pp. 401-422 Mar. 2004*.
- [54] T. G. Kolda and B. W. Bader. Tensor decompositions and applications. *Society for Industrial and Applied Mathematics, vol. 51, pp. 455-500, 2009*.
- [55] E. G. Larsson. Mimo detection methods: How they work [lecture notes]. *Signal Processing Magazine, IEEE, vol.26, no.3, pp.91-95, May 2009*.
- [56] E. G. Larsson, O. Edfors, F. Tufvesson, and T. Marzetta. Massive MIMO for Next Generation Wireless Systems. *IEEE Communications Magazine, vol.52, no.2, pp. 186-195, Feb. 2014*.
- [57] L. De Lathauwer and J. Castaing. Blind identification of underdetermined mixtures by simultaneous matrix diagonalization. *IEEE Transactions on Signal Processing, vol. 56, no. 3, March 2008*.
- [58] J.-H. Lee and C.-C. Wang. Adaptive array beamforming with robust capabilities under random sensor position errors. *Proceedings of IEEE Radar Sonar Navigation, vol. 152, no. 6, Dec. 2005*.
- [59] F. Li, H. Liu, and R. J. Vaccaro. Performance analysis for doa estimation algorithms: Unification, simplifications, observations. *IEEE Trans. Aerosp. Electron. Syst. , vol. 29, no. 4, pp. 1170-1184, Oct. 1993*.
- [60] J. Li, P. Stoica, and Z. Wang. On robust capon beamforming and diagonal loading. *IEEE Transactions on Signal Processing, vol. 51, no. 7, July, 2003*.
- [61] Z. Linrang, B. Zheng, and L. Guisheng. Adaptive beamforming in colored noise environment. *in Proc. International Conference on Signal Processing Proceedings, ICSP '98, pp. 343-346, 1998*.
- [62] W. Liu, S. Ding, M. Jin, and J. Kim. A robust doa/beamforming algorithm using the constant modulus feature. *International Journal of Innovative Computing, Information and Control, vol. 8, no. 3(B), March 2012*.

-
- [63] W. Liu and D. P. Mandic. Semi-blind source separation for convoltive mixtures based on frequency invariant transformation. *In Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2005)* , 2005.
 - [64] W. Liu and S. Weiss. Wideband beamforming concepts and techniques. *Wiley Series on Wireless Communications and Mobile Computing*, Wiley, 2010.
 - [65] W. Liu, R. B. Wu, and R. Langley. Analysis and a novel design of the beamspace broadband adaptive array. *in Proc. Progress in Electromagnetics Research Symposium, Cambrige, USA, 2006*.
 - [66] M. A. M. Marinho, F. Antreich, J. P. C. L. da Costa, and J. A. Nossek. A signal adaptive array interpolation approach with reduced transformation bias for doa estimation of highly correlated signals. *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), May 2014, Florence*.
 - [67] M. A. M. Marinho, J. P. C. L. da Costa, F. Antreich, and L. R. A. X. de Menezes. Unscented transformation based array interpolation. *in Proc. 40th International Conference on Acoustics, Speech, and Signal Processing (ICASSP), Brisbane, Australia, April 2015*.
 - [68] J .R. Mohammed. A new robust adaptive beamformer for enhancing speech corrupted with colored noise. *in Proc. International Conference on Computer Systems and Applications AICCSA 2008*, pp. 508-515, 2008.
 - [69] K. Mori, H. Arai, Y. Qian, and T. Itoh. Direct conversion receiver for digital beamforming at 8.45 ghz. *in Proc. of 2001 IEEE MTT-S International Microwave Symposium, vol. 2, pp. 1375-1378, May 2001*.
 - [70] D. Nion and N. D. Sidiropoulos. Adaptive algorithms to track the parafac decomposition of a third-order tensor. *IEEE Transactions on Signal Processing*, vol. 57, no. 6, pp. 2299 - 2310, June 2009.
 - [71] Association of Radio Industries and Businesses. Mobile communications systems for 2020 and beyond. *ARIB 2020 and Beyond Ad Hoc Group White Paper, October, 2014*.
 - [72] A. Papoulis. Probability, random variables, and stochastic processes. *3rd ed. New York: McGraw-Hill, 1991*.
 - [73] F. Roemer. Advanced algebraic concepts for efficient multi-channel signal processing,. *Ph.D. Thesis, Ilmenau University of Technology, 2013*.

- [74] F. Roemer, M. Haardt, and G. Del Galdo. Analytical performance assessment of multi-dimensional matrix- and tensor-based esprit-type algorithms. *IEEE Transactions on Signal Processing*, vol. 62, pp. 2611-2625, May 2014.
- [75] F. Rusek, D. Persson, B. K. Lau, E. G. Larsson, T. L. Marzetta, O. Edfors, and F. Tufvesson. Scaling Up MIMO: Opportunities and Challenges with Very Large Arrays. *IEEE Signal Processing Magazine*, vol.30, no.1, pp. 40-60, Jan. 2013.
- [76] T. Sekiguchi and Y. Karasawa. Wideband beamspace adaptive array utilizing fir fan filters for multibeam forming. *IEEE Transactions on Signal Processing*, vol. 48, issue 1, pp. 277 - 284, 2000.
- [77] Matteo Sgammini, Felix Antreich, Lothar Kurz, Michael Meurer, and Tobias G. Noll. Blind Adaptive Beamformer Based on Orthogonal Projections for GNSS. *Proceedings of ION GNSS*, 2012.
- [78] R. R. Silva, J. P. C. L. da Costa, R. K. Miranda, and G. Del Galdo. Applying base value of fundamental frequency via the multivariate kernel-density in forensic speaker comparison. in *Proc. 10th International Conference on Signal Processing and Communication Systems ICSPCS, Gold Coast*, 2016.
- [79] P. Strobach. Bi-iteration svd subspace tracking algorithms. *IEEE Transactions on Signal Processing*, vol. 45, pp. 1222-1240, May 1997.
- [80] J. E. Thorner. Approaches to sonar beamforming. in *Proc. of Southern Tier Technical Conference*, pp. 69-78, Apr. 1990.
- [81] C.-Y. Tseng and L. J. Griffiths. A systematic procedure for implementing the blocking. *IEEE Twenty-Second Asilomar Conference on Signals, Systems and Computers*, vol. 2, pp. 808-812, 1988.
- [82] B. D. van Veen and K. M. Buckley. Beamforming: a versatile approach to spatial filtering. *ASSP Magazine IEEE*, vol. 5 , Issue 2, pp. 4-24, Apr. 1988.
- [83] E. Vlachos and K. Berberidis. Adaptive completion of the correlation matrix in wireless sensor networks. in *Proc. 24th European Signal Processing Conference (EUSIPCO)*.
- [84] L. Wang. Array signal processing algorithms for beamforming and direction finding. *Ph.D. Dissertation, University of York*, 2009.
- [85] L. Wang, R. C. de Lamare, , and M. Yukawa. Adaptive reduced-rank constrained constant modulus algorithms based on joint iterative optimization of filters for beamforming. *IEEE Trans. Signal Processing*, vol. 58, pp. 2983-2997, Jun. 2010.

-
- [86] M. V. Woodward and P. J. Mosterman. Challenges for embedded software development. *in Proc. IEEE International Midwest Symposium on Circuits and Systems, Aug 2007.*
 - [87] Z. Yang, R.C. de Lamare, and X. Li. L1 regularized stap algorithm with a generalized sidelobe canceler architecture for airborne radar. *2011 IEEE Statistical Signal Processing Workshop, pp. 329-332, Jun. 2011.*
 - [88] L. J. Ziomek. Fundamentals of acoustic field theory and space-time signal processing. *CRC Press, 1995.*

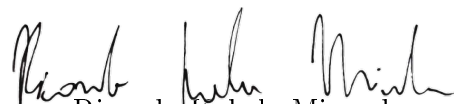
ERKLÄRUNG

Ich versichere, dass ich die vorliegende Arbeit ohne unzulässige Hilfe Dritter und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe. Die aus anderen Quellen direkt oder indirekt übernommenen Daten und Konzepte sind unter Angabe der Quelle gekennzeichnet.

Weitere Personen waren an der inhaltlich-materiellen Erstellung der vorliegenden Arbeit nicht beteiligt. Insbesondere habe ich hierfür nicht die entgeltliche Hilfe von Vermittlungs bzw. Beratungsdiensten (Promotionsberater oder anderer Personen) in Anspruch genommen. Niemand hat von mir unmittelbar oder mittelbar geldwerte Leistungen für Arbeiten erhalten, die im Zusammenhang mit dem Inhalt der vorgelegten Dissertation stehen.

Die Arbeit wurde bisher weder im In- noch im Ausland in gleicher oder ähnlicher Form einer Prüfungsbehörde vorgelegt. Ich bin darauf hingewiesen worden, dass die Unrichtigkeit der vorstehenden Erklärung als Täuschungsversuch bewertet wird und gemäß § 7 Abs. 10 der Promotionsordnungen Abbruch des Promotionsverfahrens zur Folge hat.

Brasília, der 10.04.2017


Ricardo Kehrle Miranda